

IMiS[®] /wScan
Priročnik

Verzija 1.3.1802

IMAGING
SYSTEMS

Imaging Systems, informacijski sistemi, d.o.o.
Brnčičeva 41g
Ljubljana

KAZALO VSEBINE

1	UVOD	7
1.1	O priručniku	7
1.2	Ciljno občinstvo	7
2	SPLOŠNO	7
2.1	Arhitektura	9
2.1.1	Modularna zasnova	9
2.1.2	Več-nivojska arhitektura	10
2.2	Varnost	13
2.3	Funkcionalnosti	14
2.3.1	Modul za zajem vsebin iz optičnega čitalca	14
2.3.2	Modul za shranjevanje vsebin	15
2.3.3	Modul za prepoznavo barkod	15
2.4	Integracija z aplikacijami	16
2.4.1	Integracija storitve IMiS®/Capture Service	16
2.4.2	Integracija knjižnice imis.scan.js	16
2.4.3	Integracija knjižnice imis.scan.ui.js	18
2.5	Verzioniranje	19
3	SISTEMSKE ZAHTEVE	20
3.1	Strojna oprema	20
3.1.1	Minimalne zahteve	20
3.1.2	Priporočene zahteve	20
3.2	Programska oprema	20
4	UPRAVLJANJE	21
4.1	Namestitev	21
4.1.1	Namestitev s čarovnikom	23
4.1.2	Tiha namestitev	28
4.2	Zagon in zaustavitev	31
4.3	Dodatne nastavitve	32
4.3.1	Dodatno nastavljanje profilov	34
4.3.2	Varnostne nastavitve	36
4.3.3	Administratorske dodatne nastavitve	37
4.4	Odstranitev in spremembe	37
4.4.1	Odstranitev	38
4.4.2	Spremembe in popravki v namestitvi	40
4.4.2.1	Spremembe v namestitvi	41
4.4.2.2	Popravki v namestitvi	43

4.5	Nadgradnja.....	43
5	TEHNIČNA DOKUMENTACIJA.....	44
5.1	imis.scan.js.....	44
5.1.1	imis.scan.Scan.....	44
5.1.2	imis.scan.Profile.....	48
5.1.3	imis.scan.Job.....	49
5.1.4	imis.scan.Document.....	52
5.1.5	imis.scan.Page.....	53
5.1.6	imis.scan.Barcode.....	54
5.1.7	imis.scan.Module.....	54
5.1.8	imis.scan.ScannerModule.....	55
5.1.9	imis.scan.FolderTargetModule.....	56
5.1.10	imis.scan.BarcodeExtractorModule.....	56
5.1.11	imis.scan.ScannerValue.....	58
5.1.12	imis.scan.ColorFormat.....	58
5.2	imis.scan.ui.js.....	59
5.2.1	imis.scan.ui.Scan.....	59
5.2.1.1	UIScanOptions.....	59
5.2.1.2	UIScanButtonsOptions.....	60
5.2.2	imis.scan.ui.Button.....	60
5.2.3	imis.scan.ui.ColorDropdownButton.....	62
5.2.4	imis.scan.ui.ProfilesButton.....	62
5.2.5	imis.scan.ui.ImageDetails.....	63
5.2.6	imis.scan.ui.ImageView.....	64
5.2.7	imis.scan.ui.ImageScroll.....	65
5.2.8	imis.scan.ui.Progress.....	66
5.2.9	imis.scan.ui.Status.....	67
5.2.10	imis.scan.ui.Thumbnails.....	67
5.2.11	imis.scan.ui.Settings.....	69
5.2.12	imis.scan.ui.AlertDialog.....	70
5.3	Primeri uporabe imis.scan.js.....	70
5.3.1	Branje profilov.....	70
5.3.2	Spreminjanje profila.....	72
5.3.3	Zagon opravila.....	74
5.3.4	Brisanje profila.....	77
5.4	Primeri uporabe imis.scan.ui.js.....	78
5.4.1	Klasičen način.....	79
5.4.1.1	classic.html.....	80

5.4.1.2	classic.css.....	82
5.4.2	Moderni način.....	83
5.4.2.1	modern.html.....	84
5.4.2.2	modern.css	86
5.4.3	Klasičen temni način.....	87
5.4.3.1	classic_dark.html.....	88
5.4.3.2	classic.dark.css	90
5.4.4	Galerija.....	91
5.4.4.1	gallery.html.....	92
5.4.4.2	gallery.css.....	93
6	ODPRAVLJANJE TEŽAV	94
6.1	Težave pri uporabi IMiS®/wScan	94
6.1.1	Napaka »Forbidden«	94
6.1.2	Napaka »Error in establishing connection«.....	94
6.1.3	Napaka »Socket connection error«.....	95
6.1.4	Napaka »No scanner is connected«.....	95
6.1.5	Napaka »Scanner: '{ime skenerja}' can not be loaded«.....	95
6.1.6	Po uspešnem skeniranju ni možno nadaljevati skeniranja	96
6.1.7	Med skeniranjem se prazne strani ne odstranjujejo	96
6.1.8	Napaka »Your browser does not support Javascript ES6. Update browser.«.....	96
6.1.9	Napaka »Your browser does not support WebSockets. Update browser.«.....	97

KAZALO SLIK

V nadaljevanju je uporabniku na voljo seznam slik uporabljenih v tem priručniku.

Slika 1: Shematični prikaz umestitve produkta v večnivojski spletni aplikaciji.....	8
Slika 2: Shematični prikaz jedrne storitve IMiS®/Capture Service.....	9
Slika 3: Shematični prikaz JavaScript programskih nivojev rešitve IMiS®/wScan.....	11
Slika 4: Primer uporabe knjižnice imis.scan.js za branja profilov	17
Slika 5: Primer uporabe knjižnice imis.scan.ui.js za nastavitve izgleda Thumbnails komponente	18
Slika 6: Obvestilo o nedovoljeni hkratni namestitvi 64-bitne in 32-bitne verzije IMiS®/wScan	21
Slika 7: Obvestilo o nedovoljeni hkratni namestitvi 32-bitne in 64-bitne verzije IMiS®/wScan	22
Slika 8: Obvestilo o nedovoljeni hkratni namestitvi IMiS®/wScan in IMiS®/wBatchScan	22
Slika 9: Opozorilo o zahtevani namestitvi .NET Framework 4.5.....	22
Slika 10: Pregled in potrditev licenčnih pogojev	23
Slika 11: Vpis podatkov o uporabniku aplikacije.....	24
Slika 12: Izbira med običajno, polno in uporabniku prilagojeno namestitvijo	24
Slika 13: Izbira katere bližnice se bodo ustvarile in možnosti, ki se bodo aktivirale med namestitvijo	25
Slika 14: Prikaz rezultata izbire "Začetni meni"	26
Slika 15: Zagon postopka namestitve	26
Slika 16: Prikaz vrstice napredka med postopkom nameščanja	27
Slika 17: Obvestilo o zaključku postopka namestitve	27
Slika 18: Izbira elementov nameščanja aplikacije	28
Slika 19: Prikaz ukazne vrstice za tiho namestitev.....	29
Slika 20: Prikaz trenutnega stanja IMiS®/Capture Service: zaustavljen	31
Slika 21: Izbira opcije za zagon IMiS®/Capture Service.....	31
Slika 22: Prikaz stanja IMiS®/Capture Service: v delovanju	31
Slika 23: Izbira opcije za zaustavitev IMiS®/Capture Service	32
Slika 24: Izbira opcije za ponovni zagon IMiS®/Capture Service	32
Slika 25: Izbira opcije za prikaz dodatnih nastavitvev	33
Slika 26: Pogovorno okno za nastavljanje dodatnih nastavitvev.....	33
Slika 27: Pogovorno okno za ponovni zagon storitve IMiS®/Capture Service	34
Slika 28: Pogovorno okno za nastavitve profilov in varnostne nastavitve	35
Slika 29: Nastavitveno okno Fujitsu PapierStream gonilnika	35
Slika 30: Pogovorno okno za varnostne nastavitve	36
Slika 31: Izbira med spremembo v namestitvi in odstranitvijo aplikacije.....	38
Slika 32: Izbira odstranitve aplikacije.....	38
Slika 33: Prikaz vrstice napredka pregleda konfiguracije	38
Slika 34: Izbira odstranitve aplikacije.....	39

Slika 35: Potrditev odstranitve aplikacije	39
Slika 36: Prikaz vrstice napredka med postopkom odstranitve aplikacije	40
Slika 37: Obvestilo o zaključku postopka odstranitve namestitvenega paketa.....	40
Slika 38: Izbira med spremembo in popravki v namestitvi ter odstranitvijo nameščene aplikacije .	41
Slika 39: Zagon postopka izvedbe sprememb in popravkov v namestitvi.....	41
Slika 40: Izbira spremembe namestitve	42
Slika 41: Izbira elementov pri spremembi namestitve.....	42
Slika 42: Izbira popravkov namestitve.....	43
Slika 43: Komponenta gumb.....	60
Slika 44: Komponenta za izbiro barve skeniranja.....	62
Slika 45: Komponenta za izbiro profila in spreminjanje nastavitev profila za skeniranje.....	62
Slika 46: Komponenta za prikaz informacij trenutno izbrane strani.....	63
Slika 47: Komponenta za prikaz trenutno izbrane strani	64
Slika 48: Komponenta za prikaz zbirke strani	65
Slika 49: Komponenta za prikaz trenutnega opravila.....	66
Slika 50: Komponenta za prikaz statusa	67
Slika 51: Komponenta za prikaz dokumentov	68
Slika 52: Komponenta za nastavljanje profilov.....	69
Slika 53: Komponenta za prikaz dialoga.....	70
Slika 54: Primer uporabe klasičnega načina prikaza uporabniškega vmesnika	79
Slika 55: Primer uporabe modernega načina prikaza uporabniškega vmesnika	83
Slika 56: Primer uporabe klasičnega (temnega) načina prikaza uporabniškega vmesnika	87
Slika 57: Primer uporabe galerija načina prikaza uporabniškega vmesnika	91

1 UVOD

1.1 O priručniku

Priručnik IMiS®/wScan opisuje funkcionalnosti in način dela z aplikacijo IMiS®/wScan.

1.2 Ciljno občinstvo

Namenjen je administratorjem in razvijalcem aplikacij s tehničnim predznanjem, ki potrebujejo informacije o namestitvi in konfiguraciji IMiS®/wScan aplikacije, njenem jedru

IMiS®/CaptureService ter integraciji z aplikacijami različnih ponudnikov.

Razvijalcem aplikacij je v nadaljevanju na voljo tehnična dokumentacija s podrobnim opisom storitve IMiS®/Capture Service.

2 SPLOŠNO

IMiS®/wScan sledi vsem modernim tehnološkim, funkcionalnim in oblikovnim standardom na področju programske opreme za zajem fizičnih dokumentov. V svoji zasnovi omogoča polno funkcionalno uporabo v več nivojski arhitekturi, kjer je točka integracije spletni brskalnik. Zaradi svoje modularne in nivojske zasnove je visoko prilagodljiv in uporaben v različnih implementacijskih scenarijih, bodisi z ali brez uporabniške interakcije.

Kljub temu, da je za zajem fizičnega gradiva skoraj nujna uporaba fizičnih računalniških komponent (optični čitalec) katerih uporabo so proizvajalci spletnih brskalnikov praktično onemogočili, IMiS®/wScan s uporabo inherentno varnih tehnologij kljub temu omogoča digitalizacijo fizičnih dokumentov v čisti spletni rešitvi brez uporabe vtičnikov ali podobnih komponent.

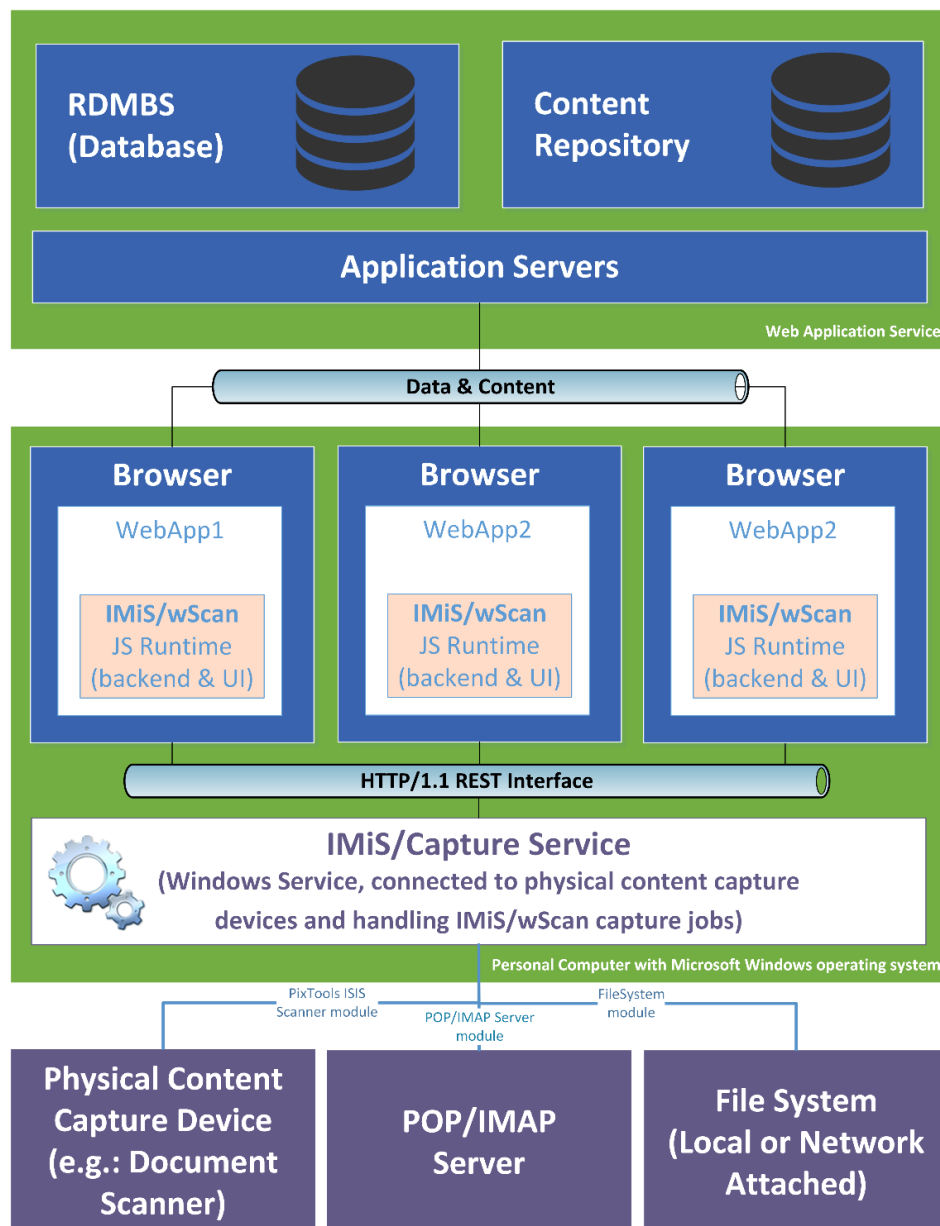
Uporabnikom omogoča zajem vsebin in njihovo digitalizacijo v čistih spletnih rešitvah.

Izdelan je na osnovi specifikacije ECMAScript 2016. Kljub precej novi specifikaciji JavaScript jezika, je podpora v brskalnikih zadostno zagotovljena.

Spletna storitev konceptualno zadošča naslednjim kriterijem:

- zasnovan je na čisti JavaScript tehnologiji, brez dodatnih zahtev po npr. vtičnikih ali dostopu do »nativnih« protokolov NPAPI, COM, Potrebno je poudariti, da sloni na tehnologijah, ki jih proizvajalci brskalnikov obravnavajo kot neustrezne (nevarne);

- enostaven, intuitiven in prilagodljiv uporabniški vmesnik omogoča razvijalcem aplikacij popolno prilagodljivost pri integraciji v poljubne spletne aplikacije;
- integracija na dovolj nizkem nivoju omogoča razvijalcem aplikacij prilagoditve tudi v primeru tehnoloških konfliktov s kakšnim interno uporabljenim ogrodjem v primeru, ko razvijalec aplikacij ne more uporabiti npr. konstruktov View nivoja aplikacije.

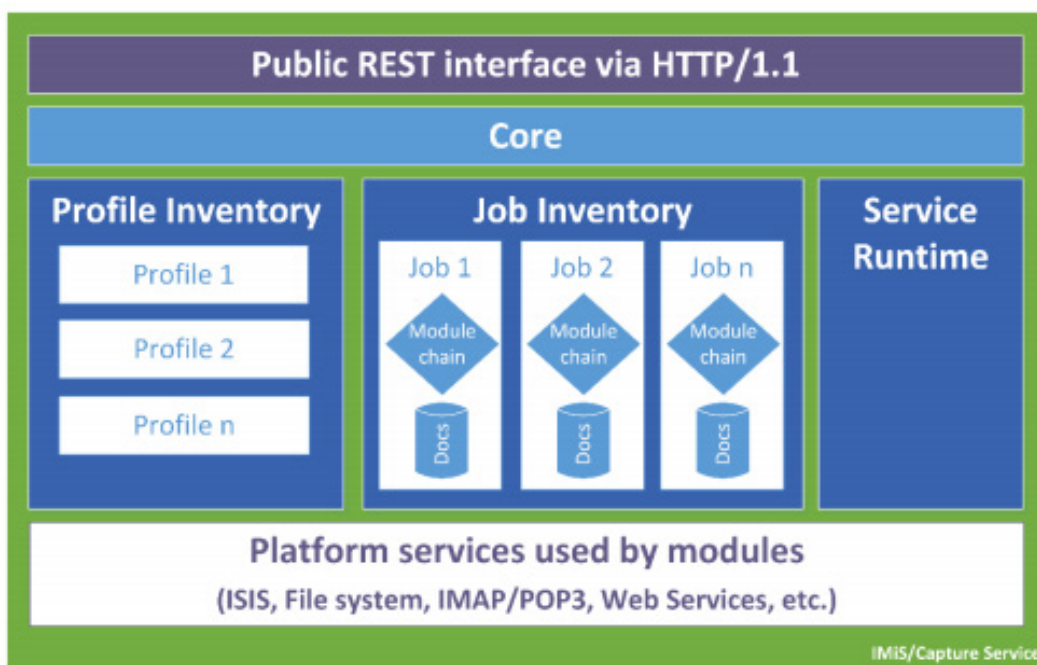


Slika 1: Shematični prikaz umestitve produkta v večnivojski spletni aplikaciji

2.1 Arhitektura

2.1.1 Modularna zasnova

IMiS®/Capture Service je zasnovan na modularno veriženi zasnovi, kjer lahko glede na potrebe opravila nizamo različne monolitne gradnike (module). Vsak modul nudi določeno funkcionalnost (npr. modul za komunikacijo z optičnim čitalcem, modul za prepoznavo črtne kode, modul za združevanje dokumentov, modul za združevanje dokumentov, modul za pretvorbo dokumentov, modul za shranjevanje na arhivski sistem, ... itd.). Modularna zasnova omogoča zajem iz različnega in heterogenega nabora virov.



Slika 2: Shematični prikaz jedrne storitve IMiS®/Capture Service

***Opomba:** Nadaljnji razvoj predvideva postopno uvedbo modulov za zajem dokumentov iz poštnih nabiralnikov preko IMAP/POP3 tehnologije, datotečnega sistema, zunanjih virov preko različnih spletnih storitev, ... itd. Ključno je, da bo to enotna točka za zajem in procesiranje (ločevanje logičnih dokumentov po črtnih kodah, ipd) vseh vhodnih dokumentov ne glede na to, po kakšnem kanalu so prišle do zajema.*

2.1.2 Več-nivojska arhitektura

IMiS®/Capture Service je Windows kompatibilna zaledna storitev za zajem dokumentov in krmiljenje priključenih ISIS kompatibilnih optičnih čitalcev. Za te funkcije uporablja tehnologijo Captiva PixTools (<http://documentum.opentext.com/captiva-oem/software/pixtools-toolkit/>) v svoji Microsoft .NET implementaciji. Da bi omogočili kar se da širok nabor možnosti za integracijo v različne tehnologije, je storitev zasnovana brez uporabniškega vmesnika.

Njegovo funkcionalnost v popolnosti izrabljamo preko njegovega REST vmesnika. Ta je dostopen preko .NET spletnega strežnika vgrajenega v storitev in deluje na osnovi tehnologije Hypertext Transfer Protocol -- HTTP/1.1 (<https://www.w3.org/Protocols/rfc2616/rfc2616.html>).

Funkcionalno nudi storitev upravljanja z življenjskim ciklom:

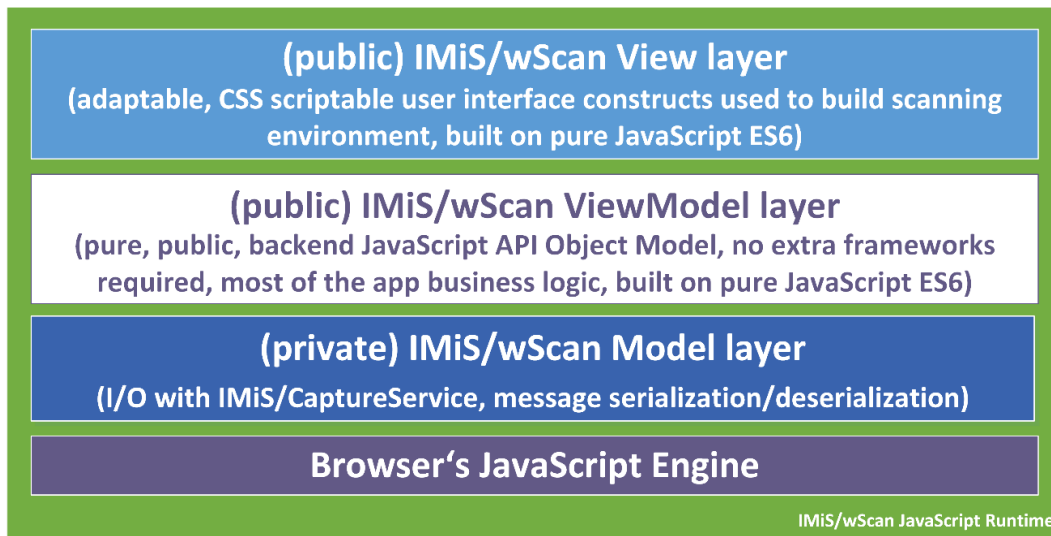
- nastavitve za zajem (t. i. Profile Lifecycle), ki jih lahko uporabljamo pri zajemu (profil je shranjen skupek nastavitvev opravila za zajem dokumentov);
- opravi zajema (t. i. Job Lifecycle);
- posameznega zajetega dokumenta (t. i. Document Lifecycle).

Programski vmesnik produkta IMiS®/wScan je enostaven, intuitiven in zelo prilagodljiv.

Razvijalcem aplikacij omogoča popolno prilagodljivost funkcionalnosti poljubnim spletnim aplikacijam. Zasnovan je več-nivojsko, na čisti JavaScript tehnologiji brez zahtev po zunanjih JavaScript ogrodjih (npr. vtičnikih ali dostopu do »nativnih« protokolov NPAPI, COM, ...).

Z integracijo se lahko razvijalci spustijo na dovolj nizek nivo, ki omogoča vse potrebne prilagoditve.

To pride v poštev zlasti, ko uporabnik zaradi tehnoloških konfliktov s kakšnim interno uporabljenim ogrodjem ne more uporabiti na primer konstruktov View (UI) nivoja aplikacije.



Slika 3: Shematični prikaz JavaScript programskih nivojev rešitve IMiS®/wScan

JavaScript programska komponenta brskalnika (angl. Browser's JavaScript Engine) je prvi in najosnovnejši nivo za komunikacijo s fizičnimi komponentami sistema.

Model nivo (angl. Model layer) aplikacije je privaten del aplikacije in ni namenjen točkam integracije z aplikacijami. Skrbi za povezavo z IMiS®/Capture Service preko HTTP povezave (navadno lokalna, lahko je tudi oddaljena) in nudi podporo ViewModel nivoju. Navzven izpostavlja notranji (zaseben) objektni model, kjer so vsa sporočila storitve že deserializirana in na voljo višjim nivojem v obliki JavaScript objektov. Prav tako skrbi za asinhrono proženje dogodkov, ki izvirajo iz dogodkov na storitvenem nivoju in jih višji nivoji ne bi mogli zaznati ter se na njih odzivati. Preko tega nivoja se izmenjujejo vsi podatki in ukazi za operacije s storitvenim delom. Kljub temu, da je ta nivo razglašen za zasebnega, je koda odprta in na voljo zunanjim razvijalcem v vpogled predvsem v času razvoja aplikacije, kjer lahko napakam sledijo tudi v ta nivo.

ViewModel (angl. ViewModel layer) nivo aplikacije je srce aplikacije. Višjim nivojem (in opcijsko razvijalcem aplikacij) izpostavlja bogat in intuitiven JavaScript objektni model z vso poslovno logiko, ki skrbi za konsistentnost JavaScript objektov in stanja storitve, s katero komunicira. Gre za popolnoma zaledni nivo, brez konstruktov uporabniškega vmesnika. S svojimi objekti aplikacijam omogoča upravljanje s storitvijo IMiS®/Capture Service, upravljanje z življenjskim ciklom zajetih dokumentov, ... itd.

S svojim modelom dogodkov (angl. events) omogoča usklajevanje in sinhronizacijo dogodkov, ki izvirajo v sami aplikaciji IMiS®/wScan ali v storitvi IMiS®/Capture Service.

Za scenarije uporabe zajema, kjer uporabniški vmesnik ni zaželen ali potreben, je nivo zasnovan tako, da je z njim možno delati brez uporabniškega vmesnika na način, ki ne omejuje njegovega nabora funkcionalnosti. Gre za bolj izjemne dogodke katere arhitektura omogoča.

View nivo aplikacije kot zadnji v sklopu obsega nabor konstruktov uporabniškega vmesnika, ki zaokrožujejo aplikacijo za zajem in digitalizacijo fizičnih dokumentov. Povezuje se z nivojem ViewModel. Razvijalcem aplikacij omogoča enostavno in prilagodljivo vključitev konstruktov v aplikacijo brez natančnega poznavanja dogodkov in objektov, ki konstruktom omogočajo njegovo funkcijo. Osnoven izgled konstruktov je možno prilagajati preko njihovega objektnega modela (lastnosti) ali preko CSS stilov, s katerimi lahko razvijalec do podrobnosti prilagodi konstrukte željam in zahtevam aplikacij. Podobno kot ostali nivoji, je tudi ta zgrajen brez zahtev po kakšnih dodatnih JavaScript ogrodjih (npr AngularJS ipd.), zato njegova integracija ne povzroča konfliktov z aplikacijami.

Konstrukti View modela so konceptualno in funkcijsko medsebojno neodvisni, vendar so zaradi konsistentnosti prikazane informacije med seboj povezani preko mreže dogodkov, ki se posredujejo iz/v ViewModel nivo. Ti skrbijo za njihovo usklajenost v smislu njihove vsebine in stanja (primer prikaza napredka skeniranja).

Dogodki nastajanja novih strani v nekem dokumentu izvirajo iz storitve, ki zajema dokument.

Nastanek vsake strani se mora kot dogodek propagirati do vizualnih kontrol. Te so preko modela dogodkov povezani z ViewModel nivojem, ta je povezan v Model nivojem, ta pa asinhrono z WebSocket tehnologijo s storitvenim nivojem (saj bi sicer moral intervalno izvajati zahteve do storitve, v kakšnem stanju je). Tak dogodek se iz storitvenega nivoja propagira do Model nivoja odjemalca, ta ga posreduje ViewModel nivoju, ta pa osveži vse vizualne konstrukte, ki so naročeni na tak dogodek. Te so sprogramirane, da tako novo stran s storitve priključijo in jo dodajo seznamu strani, ki ga prikazujejo.

2.2 Varnost

Osnovna namestitvev ne predvideva šifriranja spletnega prometa REST vmesnika, saj promet poteka lokalno preko lokalnega omrežnega vmesnika »localhost«, tako da zaščita načeloma ni potrebna. V nastavitvah je prilagoditev možna, vendar zahteva poglobljena znanja in pravilne nastavitve uporabniških pravic. Storitvev za zajem vsebin IMiS®/Capture Service privzeto posluša na omrežnem vmesniku »localhost« (127.0.0.1 oz. ::1) na vratih 5000/tcp, kar omogoča lokalno komunikacijo z aplikacijo IMiS®/wScan. Dodatna avtentikacija uporabnika zato ni potrebna.

Vsak dostop do IMiS®/Capture Service mora vsebovati poseben niz znakov (varnostni ključ), ki mora biti zapisan v nastavitvah storitve IMiS®/Capture Service. Varnostni ključ lahko uporabnik s administratorskimi pooblastili vnese ali ga ustvari preko IMiS®/wScan administrativnega modula.

Dostop do storitve IMiS®/Capture Service je zaščiten s C.O.R.S. standardom (https://en.wikipedia.org/wiki/Cross-origin_resource_sharing), ki preprečuje spletnim brskalnikom dostop do storitev iz spletnih domen, ki niso določene v sami storitvi IMiS®/Capture Service. Spletni brsklanik bo preprečil dostop do lokalne storitve IMiS®/Capture Service vsaki spletni aplikaciji, ki se nahaja na drugi spletni domeni, v kolikor ta domena ni dovoljena v storitvi IMiS®/Capture Service. Uporabnik z administratorskimi pooblastili vnese vse dovoljene spletne domene preko IMiS®/wScan administrativnega modula.

Uporabnik, prijavljen v operacijski sistem Windows, lahko preko aplikacije IMiS®/wScan dostopa do storitev, ki jih podpira IMiS®/Capture Service. Storitvev IMiS®/Capture Service se izvaja v kontekstu systemskega uporabnika (račun SYSTEM), kar mu omogoča večji dostop do virov operacijskega sistema, kot pa navadnemu uporabniku z omejenimi pravicami (restricted user). Samo preko storitve IMiS®/Capture Service in aplikacije IMiS®/wScan so določeni viri na voljo tudi navadnemu uporabniku.

Vse nastavitve storitve IMiS®/Capture Service so shranjene v Windows registru ali na datotečnem sistemu, do katerih uporabnik brez administratorskih pravic nima dostopa.

Enako velja za uporabniško nastavljene profile skeniranja. Nastavitve profilov je možno spreminjati samo z aplikacijo IMiS®/wScan ali z administratorskim modulom storitve IMiS®/Capture Service.

Uporabnik z administratorskimi pooblastili jih lahko spreminja tudi mimo teh dveh produktov, vendar mora imeti ustrezno znanje za spreminjanje Windows registra.

Za več informacij glej [poglavje 4.3.1 Administratorske dodatne nastavitve](#).

2.3 Funkcionalnosti

- Zajem vsebin in krmiljenje priključenih ISIS kompatibilnih optičnih čitalcev.
- Zajem vsebin preko različnih spletnih brskalnikov (npr. Google Chrome, Mozilla Firefox, Microsoft Edge). Namenska aplikacija za skeniranje zato ni potrebna.
- Aplikacijo IMiS®/wScan lahko vključimo v obstoječe spletne aplikacije.
- Celoten zajem vsebine in njene obdelave se izvajajo v zaledni storitvi IMiS®/Capture Service, ki sloni na Microsoft .NET tehnologiji.
- Storitev IMiS®/Capture Service vsebuje module za zajem in obdelavo vsebin (prepoznavo barkod, zajem metapodatkov, ipd.).
- Storitev IMiS®/Capture Service je zasnovana modularno, kjer je vsak modul odgovoren za svojo fazo zajema oz. obdelave vsebine. To omogoča enostavno in hitro nadgradnjo dodatnih modulov. Z ažuriranjem profila je možno sestaviti lastno zaporedje izvajanja modulov.
- Aplikacija IMiS®/wScan je zasnovana v JavaScript jeziku, brez uporabe dodatnih tehnologij, ki jih proizvajalci brskalnikov obravnavajo kot neustrezne, oziroma nevarne (npr. ActiveX vtičniki ali dostop do »nativnih« protokolov NPAPI, COM, ...).
- IMiS®/wScan knjižnice omogočajo fleksibilnost, prilagodljivost, enostaven in hiter razvoj lastnih spletnih rešitev z uporabo JavaScript jezika.

2.3.1 Modul za zajem vsebin iz optičnega čitalca

Storitev IMiS®/Capture Service lahko zajema vsebine z vseh skenerjev, ki podpirajo industrijski standard ISIS. Standard omogoča bogat nabor funkcionalnosti in je podprt s strani večine izdelovalcev skenerjev.

Preko IMiS®/wScan aplikacije je možno spreminjati standardne nastavitve skenerja:

- izbira skenerja
- način skeniranja
- resolucijo skeniranja
- velikost skenirane strani.

Dodatne nastavitve, ki so specifične za posamezen skener, ni možno nastaviti preko IMiS®/wScan aplikacije, temveč samo preko administratorskega modula storitve IMiS®/Capture Service.

Za več informacij glej [poglavje 4.3.1 Administratorske dodatne nastavitve](#).

2.3.2 Modul za shranjevanje vsebin

Storitev IMiS®/Capture Service omogoča shranjevanje vsebin na datotečni sistem.

Na voljo so različni formati zapisa datotek:

- BMP
- GIF
- TIFF
- JPEG
- PCX
- PDF/A
- PNG.

Za vsak format shranjevanja je možno nastaviti tudi barvo in kompresijo, ki jih izbrani format datoteke podpira.

2.3.3 Modul za prepoznavo barkod

Storitev IMiS®/Capture Service omogoča prepoznavo naslednjih barkod:

- 1D barkode:
Addon 2, Addon5, Australian Post, BDC Matrix, Codabar, Code-25 Datalogic, Code-25 IATA, Code-25 Industrial, Code-25 Interleaved, Code-25 Invert, Code-25 Matrix, Code-32, Code-39, Code-93, EAN-13, EAN-8, Type-128, UCC-128, UPC-A, UPC-E,
- 2D barkode:
AZTEC, Data Matrix, Intelligent Mail, PDF-417, Postnet, QR Code, Royal Post.

2.4 Integracija z aplikacijami

Aplikacija IMiS®/wScan sestavljajo trije moduli:

- **IMiS®/Capture Service:** zaledna Windows storitev, ki izvaja zajem in obdelavo različnih vsebin;
- **imis.scan.js:** Javascript knjižnica, ki omogoča komunikacijo s storitvijo IMiS®/Capture Service;
- **imis.scan.ui.js:** pomožna JavaScript knjižnica za prikaz že narejenih vizualnih komponent.

Vsakega od navedenih modulov je mogoče uporabiti pri integraciji z drugo aplikacijo.

2.4.1 Integracija storitve IMiS®/Capture Service

Neposredna integracija s storitvijo IMiS®/Capture Service na osnovi RESTful tehnologije zaenkrat ni predmet tega priročnika.

2.4.2 Integracija knjižnice imis.scan.js

Knjižnica »imis.scan.js« skrbi za izmenjavo podatkov s storitvijo IMiS®/Capture Service v JSON formatu preko RESTful tehnologije. Preko nje lahko razvijalec aplikacij v Javascript jeziku nastavlja profile ali izvaja zajem vsebine (npr. skeniranje). Omogoča zaznavo dogodkov med skeniranjem, branje opravi, dokumentov in strani. Knjižnica je obenem osnova za izdelavo lastnih spletnih rešitev. Za svoje delovanje ne potrebuje nikakršnih drugih Javascript knjižnic.

Razvijalec mora pridobiti enolični varnostni ključ, ki je zapisan v storitvi IMiS®/Capture Service. Za pridobitev varnostnega ključa glej [poglavje 4.3.2 Varnostne nastavitve](#).


```
<!DOCTYPE html>
<html>
<head>
  <title>imis.scan.js</title>
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="sample.css" />
</head>
<body class="sample">
  <h1>Sample</h1>
  <p>This example demonstrates reading scan profiles. Profiles are displayed in list.</p>

  <div>Profiles:</div>
  <ol id="profiles"></ol>
  <div id="error"></div>

  <script src="../imis.scan.js"></script>
  <script>
    window.addEventListener('load', function () {
      try {
        // Profiles ordered list
        var ol = document.getElementById("profiles");

        // Create a scan object
        var scan = new imis.scan.Scan();

        // Read profiles
        scan.getProfiles({
          success: function (profiles) {
            for (var i = 0; i < profiles.length; i++) {
              // Add profile to ordered list
              var li = document.createElement("li");
              li.innerHTML = profiles[i].name;
              ol.appendChild(li);
            }
          },
          error: function (error) {
            // Show error
            document.getElementById("error").innerHTML = error;
          }
        });
      } catch (e) {
        // Show error
        document.getElementById("error").innerHTML = e;
      }
    });
  </script>
</body>
</html>
```

Slika 4: Primer uporabe knjižnice imis.scan.js za branja profilov

2.4.3 Integracija knjižnice imis.scan.ui.js

Knjižnica »imis.scan.ui.js« je namenjena hitrejšemu in enostavnejšemu razvoju lastnih rešitev. Vsebuje nekaj najbolj pogosto uporabljenih vizualnih komponent (za izvajanje, zaustavitev in nadaljevanje opravil, izbiro in nastavljanje lastnosti profila, prikaz poteka opravil, prikaz izbrane strani in njenih podrobnosti ter prikaz vseh zajetih strani). Komponente lahko razvijalec spletnih aplikacij enostavno vgradi v svojo aplikacijo in si naredi uporaben uporabniški vmesnik brez naprednega znanja HTML ali CSS jezika. Vse komponente imajo na voljo določene nastavitve, preko katerih jim lahko spreminja izgled. Za delovanje potrebujejo le knjižnico »imis.scan.js« in enoličen varnostni ključ, ki je zapisan v storitvi IMiS®/Capture Service. Za pridobitev varnostnega ključa glej [poglavje 4.3.2 Varnostne nastavitve](#).

```
<div id="imis-progress"></div>
<div class="main" id="main">
  <div id="thumbnails">Thumbnails</div>
</div>
<script src="imis.scan.js"></script>
<script src="imis.scan.ui.js"></script>
</script>
window.addEventListener('load', function () {

  // Set scan version to title attribute
  document.getElementById("title").setAttribute("title", imis.scan.ui.version);

  try {
    const scan = new imis.scan.ui.Scan({
      //url: "http://example.com",
      thumbnails: new imis.scan.ui.Thumbnails({
        id: "thumbnails",
        //darkMode: false,
        orientation: "horizontal",
        thumbnail: {
          height: 200, // thumbnail height
          title: false
        },
        gallery: true,
        contextMenu: {
          enabled: false
        }
      })
    });
```

Slika 5: Primer uporabe knjižnice imis.scan.ui.js za nastavitve izgleda Thumbnails komponente

2.5 Verzioniranje

Označevanje verzij produkta temelji na shemi, ki vključuje:

- štiri ločene numerične identifikatorje (MAJOR, MINOR, RELEASE)
- identifikator 32 bitne ali 64 bitne verzije (PLATFORM).

Primer zapisa je naslednji:

IMiS.wScan.MAJOR.MINOR.RELEASE.PLATFORM.msi

Primer imena namestitvenega paketa za IMiS®/wScan:

IMiS.wScan.1.3.1802.x32.msi

Schema je sestavljena iz imena storitve IMiS®/Capture Service in naslednjih elementov:

- MAJOR: Identifikator označuje glavno različico produkta, ki predstavlja generacijo produkta. Spreminja se redko, glede na obseg izvedenih sprememb in funkcionalnosti. Sprememba nakazuje na veliko razliko v produktu glede na predhodno izdane različice. Identifikator ima nabor vrednosti od 1-n, je zvezen in se izključno povečuje.
- MINOR: Identifikator označuje manjšo različico produkta, ki se spreminja pogosteje kot glavna različica glede na obseg izvedenih sprememb, funkcionalnosti in popravkov. Sprememba manjše različice predstavlja manjše spremembe in popravke v okviru iste generacije produkta, ki jo označuje glavna verzija. Nabor vrednosti je od 1-n in ni zvezen. Z vsako spremembo MAJOR verzije se postavi na izhodišče (1).
- RELEASE: Ta identifikator označuje časovno komponento izdaje produkta po shemi »LLMM«. MM označuje mesec izdaje (nabor 01-12), LL pa zadnji dve številki leta.

Primer: izdaja produkta za februar 2018 je v RELEASE identifikatorju označena kot 1802.

- PLATFORM: Označuje na katerih operacijskih sistemih lahko aplikacijo uporabljamo. 32 bitno verzijo lahko zaženo tako na 32 bitnem operacijskem sistemu Windows, kot tudi na 64 bitnem. 64 bitno verzijo je možno zagnati samo na 64 bitnem operacijskem sistemu Windows.

3 SISTEMSKE ZAHTEVE

Aplikacija IMiS®/wScan ima za uspešno namestitev in izvajanje naslednje sistemske zahteve glede strojne in programske opreme.

3.1 Strojna oprema

Praktično vsi računalniki, ki jih lahko danes kupimo na tržišču zadoščajo strojnim zahtevam za delovanje IMiS®/wScan aplikacije.

V nadaljevanju so navedene minimalne in priporočene zahteve.

3.1.1 Minimalne zahteve

Minimalne zahteve za IMiS®/wScan:

- Intel Core 2 Duo 2 GHz procesor
- 1 GB pomnilnika
- 150 MB nezasedenega prostora na trdem disku
- dostop do omrežja preko TCP/IP protokola (IPv4 ali IPv6)

3.1.2 Priporočene zahteve

Priporočene zahteve za IMiS®/wScan:

- Intel Core i5 3 GHz procesor ali hitrejši
- 2 GB RAM pomnilnika ali več
- 250 MB nezasedenega prostora na trdem disku
- dostop do omrežja preko TCP/IP protokola (IPv4 ali IPv6)

3.2 Programska oprema

Zahteve za IMiS®/wScan:

- .NET 4.5
- Javascript ECMAScript 6

- Brskalniki z omogočeno WebSocket tehnologijo in podporo za ECMAScript6 standard.
 - Google Chrome: minimalna verzija 50;
 - Mozilla Firefox: minimalna verzija 45;
 - Microsoft Edge: minimalna verzija 20.
- Podprti operacijski sistemi:
 - Windows 10; Windows 8.x in Windows 7 SP1.

4 UPRAVLJANJE

Z aplikacijo IMiS®/wScan upravljajo administratorji in/ali razvijalci aplikacij. Upravljanje obsega področje namestitve, zagona, zaustavitve, nadgradnje in odstranitve.

4.1 Namestitev

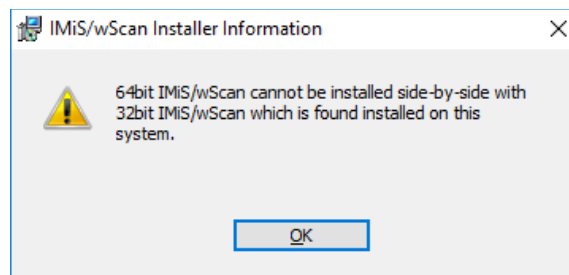
Namestitev aplikacije IMiS®/wScan se lahko izvede v okolju, ki izpolnjuje vsaj minimalne zahteve. Poteka z namestitvenim »čarovnikom«, kot administracijska namestitev ali »tiha« namestitev. V obeh primerih so vsa prikazana obvestila in pogovorna okna v angleškem jeziku.

Opozorilo:

Pred pričetkom namestitve zaustavite aplikacijo IMiS®/Scan, ker IMiS®/wScan ne deluje pravilno ob zagnani aplikaciji IMiS®/Scan.

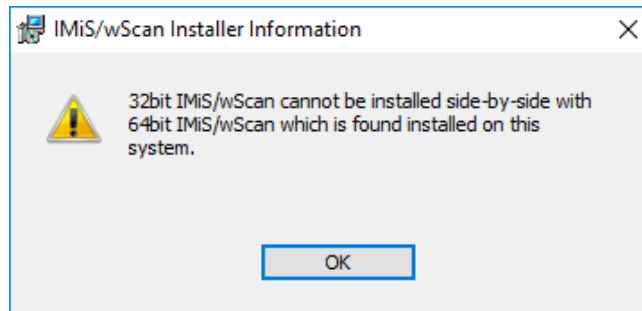
Opozorilo:

Nameščanje 64-bitne verzije IMiS®/wScan bo neuspešno v kolikor je na delovni postaji že nameščena 32-bitna verzija.



Slika 6: Obvestilo o nedovoljeni hkratni namestitvi 64-bitne in 32-bitne verzije IMiS®/wScan

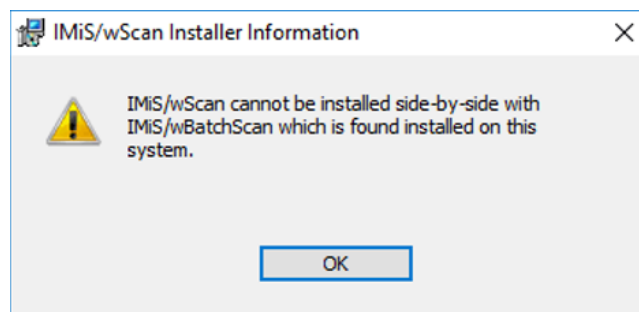
Enako velja tudi obratno. V kolikor je na delovni postaji že nameščena 64-bitna verzija IMiS®/wScan, bo nameščanje 32-bitne verzije neuspešno.



Slika 7: Obvestilo o nedovoljeni hkratni namestitvi 32-bitne in 64-bitne verzije IMiS®/wScan

Opozorilo:

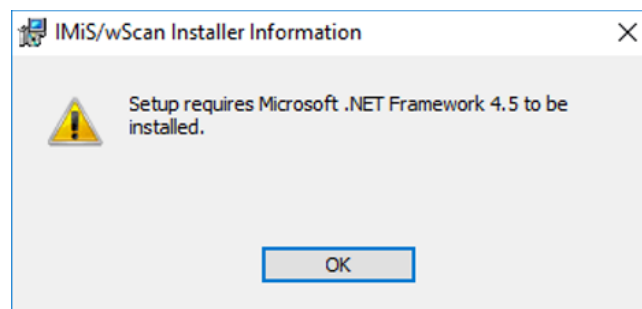
Namestitev aplikacije IMiS®/wScan na delovni postaji, kjer je že nameščen IMiS®/wBatchScan ni možno.



Slika 8: Obvestilo o nedovoljeni hkratni namestitvi IMiS®/wScan in IMiS®/wBatchScan

Opozorilo:

Namestitev aplikacije IMiS®/wScan na delovni postaji ni možno v kolikor ni nameščen .NET Framework 4.5.



Slika 9: Opozorilo o zahtevani namestitvi .NET Framework 4.5

4.1.1 Namestitev s čarovnikom

Uporabniški vmesnik namestitvenega paketa administratorja vodi skozi postopek namestitve.

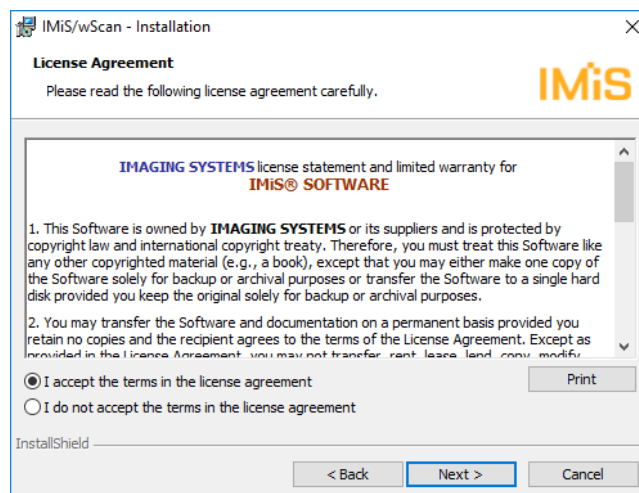
Na delovno postajo v Windows okolju s fizično priključenim enim ali več optičnimi čitalci administrator namesti aplikacijo IMiS®/wScan. Aplikacija vključuje spletno storitev IMiS®/Capture Service in ustrezne knjižnice.

Primer imena namestitvenega paketa:

IMiS.wScan.1.3.1802.x64.msi

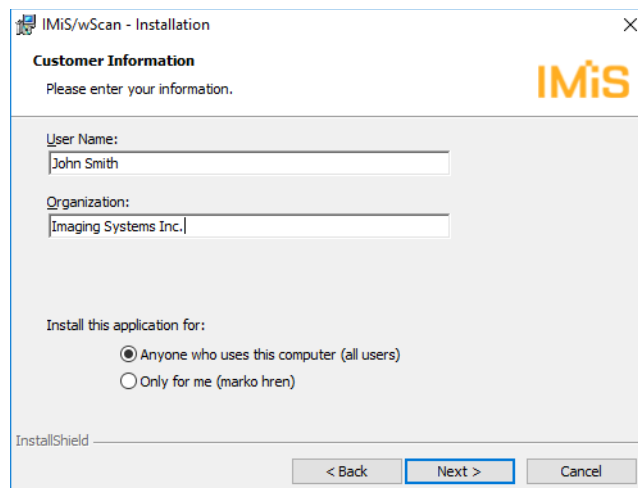
Namestitev se prične z zagonom namestitvenega paketa iz datotečnega sistema. Prikaže se pogovorno okno, ki administratorja obvesti, da se namestitveni paket pripravlja na namestitev.

V naslednjem koraku administrator pazljivo prebere določila licenčne pogodbe. V primeru, da se z njimi strinja izbere »I accept the terms in the license agreement« in s tem v celoti sprejme licenčne pogoje. V primeru, da se z licenčnimi pogoji ne strinja izbere »I do not accept terms in the license agreement« in s klikom na gumb »Cancel« prekine postopek namestitve.



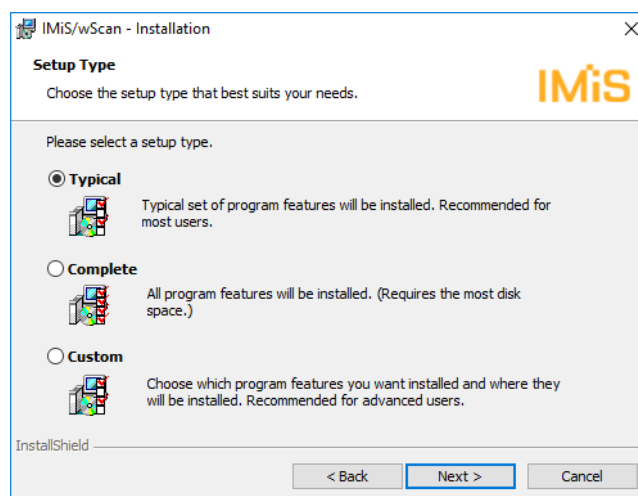
Slika 10: Pregled in potrditev licenčnih pogojev

Postopek namestitve nadaljuje z vnosom uporabniškega imena v vnosno polje »User Name« in organizacije v vnosno polje »Organization«. Izbere ali bo aplikacija nameščena samo za trenutnega uporabnika »Only for me« ali za vse uporabnike na tem računalniku »Anyone who uses this computer«.



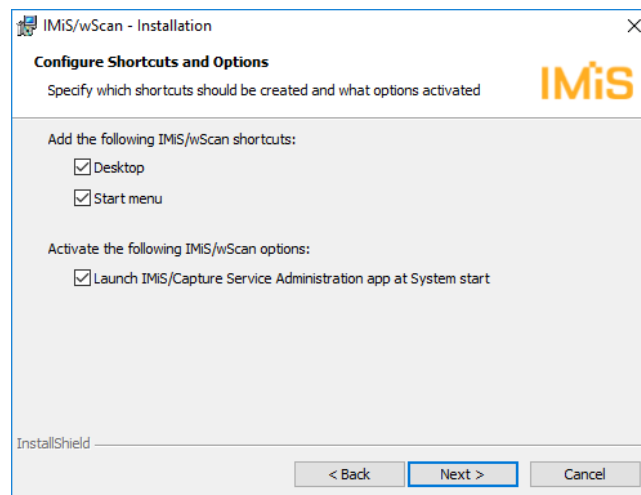
Slika 11: Vpis podatkov o uporabniku aplikacije

V naslednjem koraku izbere med običajno (angl. Typical), polno (angl. Complete) ali uporabniku prilagojeno namestitvijo (angl. Custom).



Slika 12: Izbira med običajno, polno in uporabniku prilagojeno namestitvijo

Pri vseh vrstah namestitve administrator določi katere bližnice se bodo ustvarile in katere možnosti se bodo aktivirale med postopkom nameščanja.

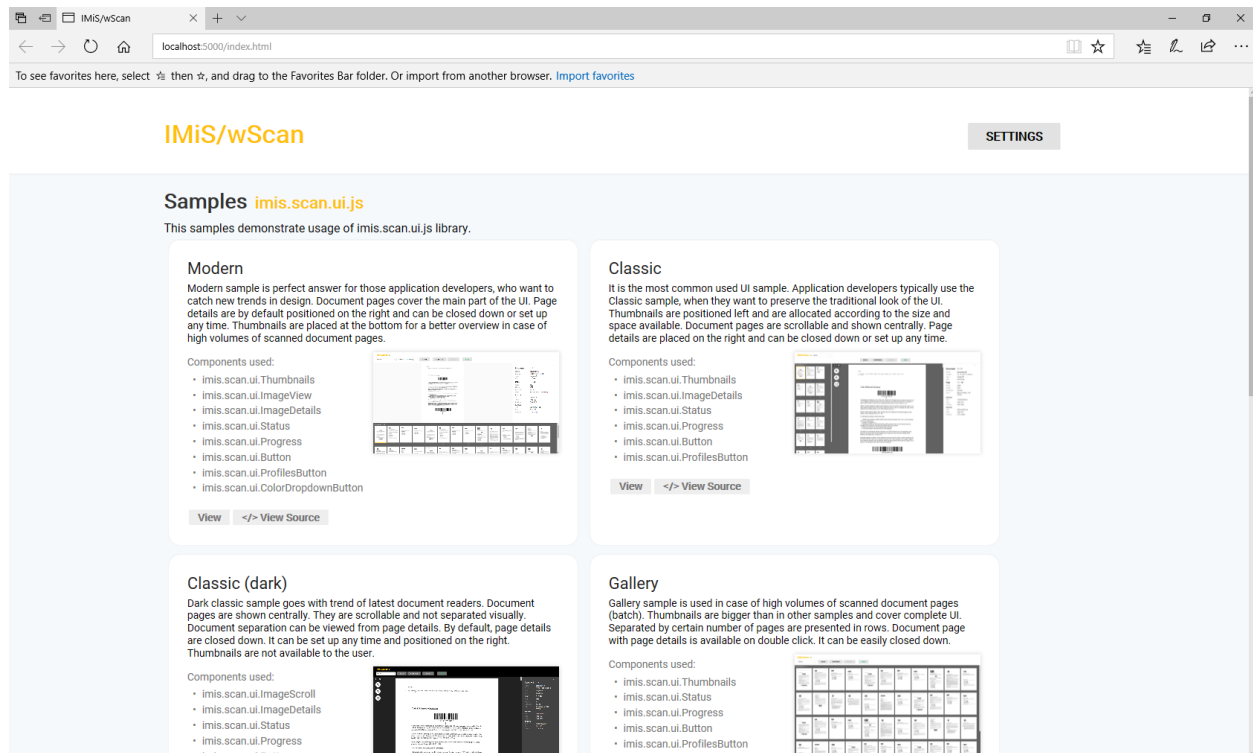


Slika 13: Izbira katere bližnice se bodo ustvarile in možnosti, ki se bodo aktivirale med namestitvijo

V kolikor administrator odkljuka izbiro »Zaženi administracijski modul storitve IMiS®/Capture Service ob zagonu delovne postaje« (angl. Launch IMiS®/Capture Service Administration app at System start) se administracijski modul zažene ob zagonu delovne postaje.

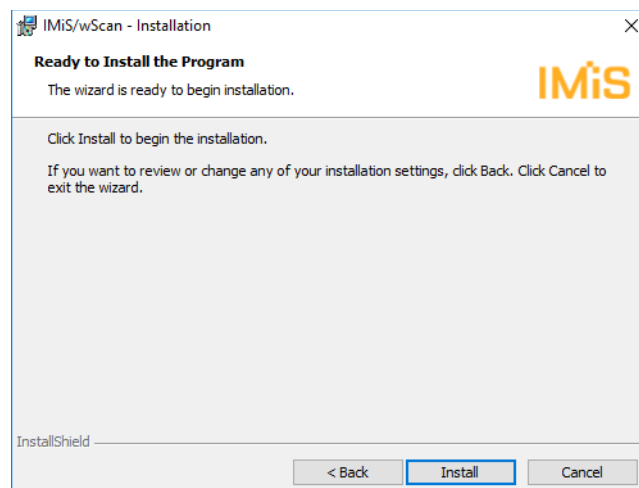
V kolikor odkljuka izbiro »Namizje« (angl. Desktop) se na namizje namestita bližnjici do IMiS®/wScan začetne strani in administratorskega modula storitve IMiS®/Capture Service.

V kolikor odkljuka izbiro »Začetni meni« (angl. Start menu) se v začetni meni doda bližnjica do IMiS®/wScan začetne strani in administratorskega modula storitve IMiS®/Capture Service.



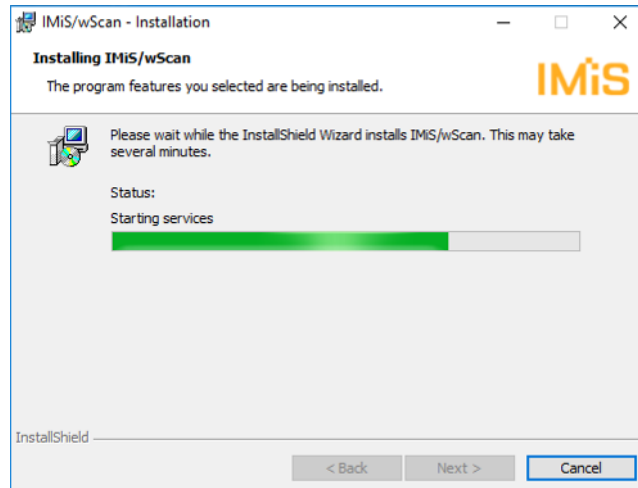
Slika 14: Prikaz rezultata izbire "Začetni meni"

Pri običajni namestitvi, ki je priporočljiva za večino uporabnikov, se izvede prenos vnaprej določenih datotek na datotečni sistem. Administrator potrdi izbrano nastavitve namestitve in zažene postopek namestitve s klikom na gumb »Install«.



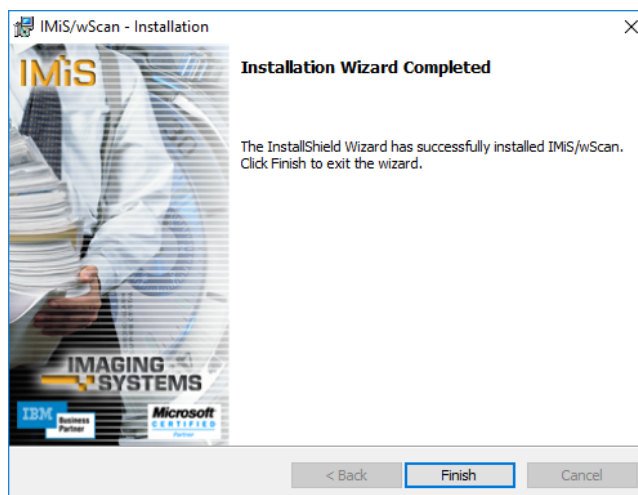
Slika 15: Zagon postopka namestitve

Prične se postopek namestitve aplikacije IMiS®/wScan. Vrstica napredka prikazuje napredek pri prenosu datotek na ustrezne lokacije. Namestitev traja nekaj deset sekund, odvisno od različice namestitvenega paketa in hitrosti računalnika.



Slika 16: Prikaz vrstice napredka med postopkom nameščanja

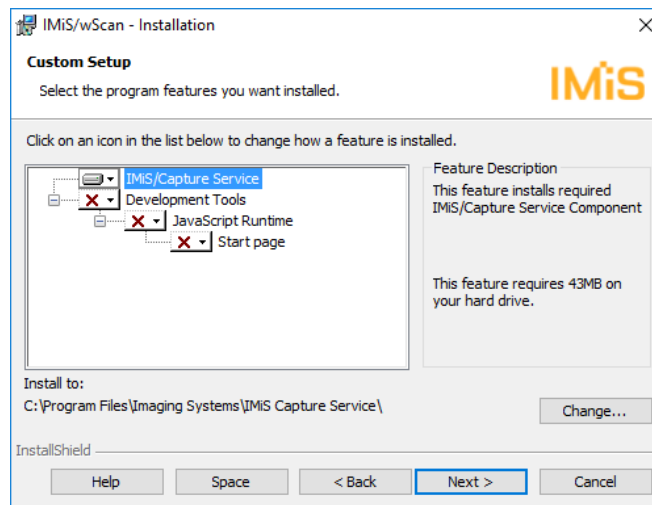
Namestitev se zaključi s prikazom zadnjega pogovornega okna, ki ga administrator zapre s klikom na gumb »Finish«.



Slika 17: Obvestilo o zaključku postopka namestitve

Enak postopek kot pri običajni namestitvi se izvede tudi pri polni namestitvi. Polna namestitev bo na datotečni sistem namestila vse elemente iz namestitvenega paketa, zato zahteva največ prostora na disku.

Uporabniku prilagojena namestitvev (angl. Custom) bo na datotečni sistem namestila samo določene elemente. Namenjena je naprednim uporabnikom.



Slika 18: Izbira elementov nameščanja aplikacije

Administrator potrdi izbrano nastavitvev namestitve in zažene postopek namestitve s klikom na gumb »*Install*«. Nadaljnji koraki so enaki kot pri običajni in polni namestitvi.

4.1.2 Tiha namestitvev

Namestitvev IMiS®/wScan aplikacije se lahko izvede tudi brez nadzora uporabnika. Namestitvev se opravi tiho, brez prikaza uporabniškega vmesnika. Za izvedbo namestitvev se uporablja pomožni program »msiexec.exe«. Ta pripomoček je del Microsoftovega namestitvenega produkta in se uporablja za opravljanje različnih vzdrževanj v aplikacijah, ki so nameščene na operacijskem sistemu Windows.

Za celoten seznam podprtih funkcij programa »msiexec.exe« glej Microsoft zbirko člankov:

[http://msdn.microsoft.com/en-us/library/windows/desktop/aa367449\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa367449(v=vs.85).aspx)

Pomožni program se izvaja iz ukazne vrstice.

Za seznam vseh parametrov glej Microsoft spletno stran:

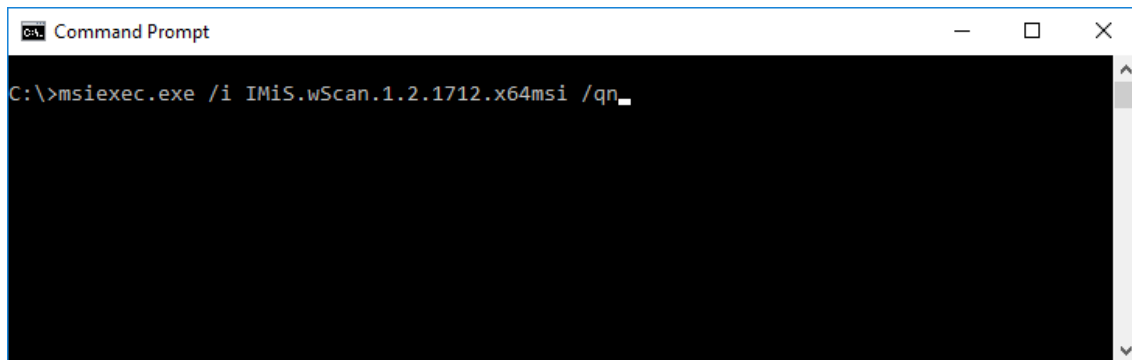
[http://msdn.microsoft.com/en-us/library/windows/desktop/aa367988\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa367988(v=vs.85).aspx).

Namestitvev lahko traja nekaj deset sekund, odvisno od hitrosti računalnika.

Primer ukazne vrstice za tiho namestitev tipične variante produkta:

C:\Windows\system32\msiexec.exe /i IMiS.wScan.1.3.1802.x64.msi /qn

V nadaljevanju je prikazana ukazna vrstica za tiho namestitev aplikacije IMiS®/wScan:



```

C:\>msiexec.exe /i IMiS.wScan.1.2.1712.x64msi /qn
  
```

Slika 19: Prikaz ukazne vrstice za tiho namestitev

V spodnji tabeli so naštetih različni načini Tihe namestitve:

Parametri ukazne vrstice	Opis
/q, /qn	Brez uporabniškega vmesnika.
/qn+	Brez uporabniškega vmesnika z modalnim oknom ob koncu namestitve.
/qb	Osnovni uporabniški vmesnik z enostavnim prikazom napredka. Za skrivanje gumba »Cancel« se uporabi parameter »/gb!«.
/qr	Poenostavljen uporabniški vmesnik brez modalnega okna ob koncu namestitve.
/qf	Celotni uporabniški vmesnik, z vsemi pogovornimi okni, prikazom napredka in napak ob koncu namestitve.

Pred začetkom namestitve IMiS®/wScan aplikacije lahko podamo različne parametre, ki so specifični za namestitev. Dodajamo jih na konec ukazne vrstice s sintakso:

```
c:\windows\system32\msiexec.exe /i IMiS.wScan.1.3.1802.x64.msi /qn PARAMETER=VALUE
```

V spodnji tabeli so opisani podprti parametri ukazne vrstice:

PARAMETER	Veljavne vrednosti	Opis
INSTALLDIR	<ime mape>	Lastnost vsebuje privzeto ponorno mapo za namestitvene datoteke. (Privzeta vrednost = "%PROGRAMFILES%\Imaging Systems\IMiS Capture Service\")
USERNAME	<uporabniško ime>	Lastnost vsebuje uporabniško ime uporabnika, ki izvaja namestitev. (Privzeta vrednost je vzeta iz sistemskih nastavitev)
COMPANYNAME	<ime podjetja>	S parametrom dodamo ime podjetja v namestitev. (Privzeta vrednost je vzeta iz sistemskih nastavitev)
SHORTCUT_START	1 / 0	S parametrom povemo namestitvenem procesu, naj naredi bližnjico v meniju »Programi«. (Privzeta vrednost = 1)
SHORTCUT_DESKTOP	1 / 0	S parametrom povemo namestitvenem procesu, naj naredi bližnjico na namizju. (Privzeta vrednost = 1)
LAUNCH_ADMIN_ON_START	1 / 0	S parametrom povemo namestitvenem procesu, naj ustvari potrebne zapise v registru in omogoči avtomatičen zagon administracijskega modula storitve IMiS®/Capture Service ob zagonu delovne postaje. (Privzeta vrednost = 1)
ADDLOCAL	ALL	Omogoči tiho namestitev vseh komponent namestitvenega paketa, kar je ekvivalentno izbiri »Complete« v namestitvi prek čarovnika.

4.2 Zagon in zaustavitev

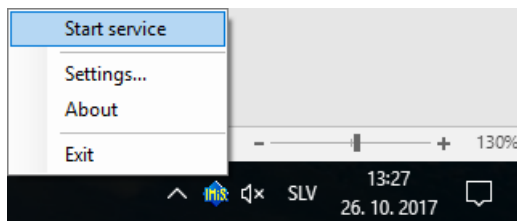
Storitev IMiS®/Capture Service se zažene samodejno ob zagonu delovne postaje povezane z optičnim čitalcem.

Zagon in zaustavitev storitve IMiS®/Capture Service je možno tudi ročno z dvoklikom miške na IMiS®/wScan administrativni modul. Po zagonu se v spodnjem delu namizja prikaže IMiS® ikona.



Slika 20: Prikaz trenutnega stanja IMiS®/Capture Service: zaustavljen

Z desnim klikom miške na IMiS® ikono se prikaže meni. Z izbiro opcije »Start service« administrator zažene storitev IMiS®/Capture Service.



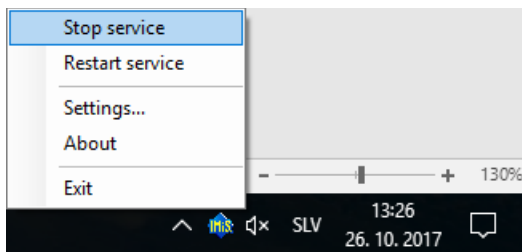
Slika 21: Izbira opcije za zagon IMiS®/Capture Service

Za zagon storitve je potrebnih nekaj sekund, da se ustrezno inicializira in preveri ustreznost gonilnika za skener. Trenutno stanje storitve je vidno s pomikom miške na IMiS® ikono.



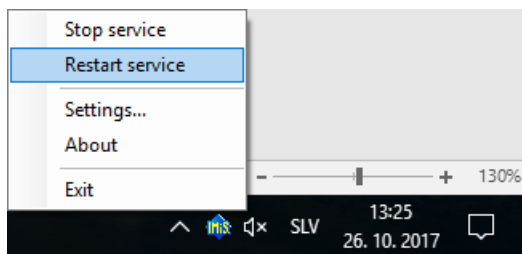
Slika 22: Prikaz stanja IMiS®/Capture Service: v delovanju

Storitev zaustavi tako, da preko menija IMiS® ikone izbere opcijo »Stop service«.



Slika 23: Izbira opcije za zaustavitev IMiS®/Capture Service

V kolikor želi administrator izvesti celotni ponovni zagon storitve, izbere opcijo »Restart service« preko menija IMiS® ikone.

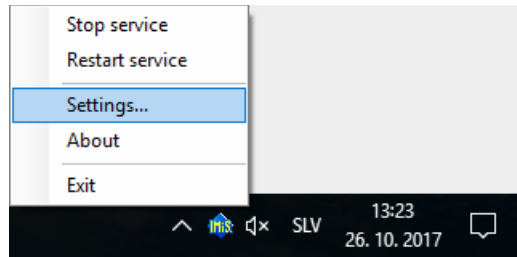


Slika 24: Izbira opcije za ponovni zagon IMiS®/Capture Service

***Opozorilo:** Uporabnik z administracijskimi pooblastili mora po ponovnem zagonu IMiS®/CaptureService obvezno osvežiti IMiS®/wScan aplikacijo v brskalniku (MS Edge, Mozilla Firefox, Google Chrome, ...).*

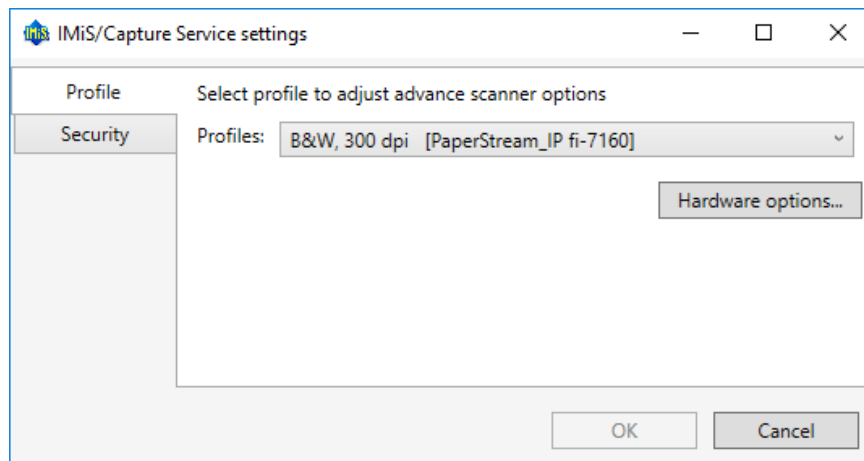
4.3 Dodatne nastavitve

Vseh nastavitvev IMiS®/wScan aplikacije ni možno izvesti preko Javascript knjižnice imis.scan.js. Dodatne nastavitve so uporabniku z administracijskimi pooblastili na voljo z dvoklikom miške na IMiS®/wScan administrativni modul. Po zagonu se v spodnjem delu namizja prikaže IMiS® ikona. Z desnim klikom miške na IMiS® ikono se prikaže meni.



Slika 25: Izbira opcije za prikaz dodatnih nastavitev

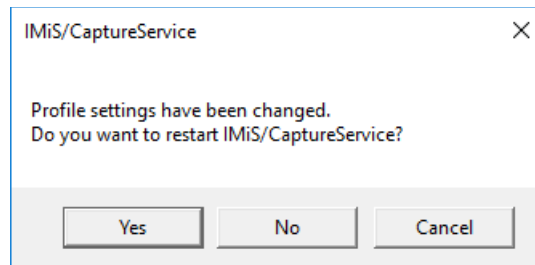
Z izbiro opcije »Settings« se prikaže nastavitveno okno.



Slika 26: Pogovorno okno za nastavljanje dodatnih nastavitev

Na voljo sta zavihka za dodatno nastavljanje profilov in varnosti. Po izbiri vsakega od zavihkov se prikažejo dodatne nastavitve. Po zaključku nastavitve uporabnik z administratorskimi pooblastili klikne na gumb »OK«.

V kolikor je bila izvedena sprememba na profilu in storitev IMiS®/Capture Service ni zagnana, se bodo dodatne nastavitve shranile. Uporabljene bodo ob naslednjem zagonu storitve IMiS®/Capture Service. V kolikor je storitev IMiS®/Capture Service zagnana, se prikaže pogovorno okno za ponovni zagon storitve IMiS®/Capture Service.



Slika 27: Pogovorno okno za ponovni zagon storitve IMiS®/Capture Service

S klikom na gumb »Yes« se prične postopek ponovnega zagona storitve IMiS®/Capture Service. Pri tem se tudi zapre nastavitveno okno.

S klikom na gumb »No« se spremenjene nastavitve samo shranijo. Upoštevale se bodo ob naslednjem zagonu storitve IMiS®/Capture Service. Nastavitveno okno se zapre.

S klikom na gumb »Cancel« se spremembe ne shranijo. Nastavitveno okno ostane odprto.

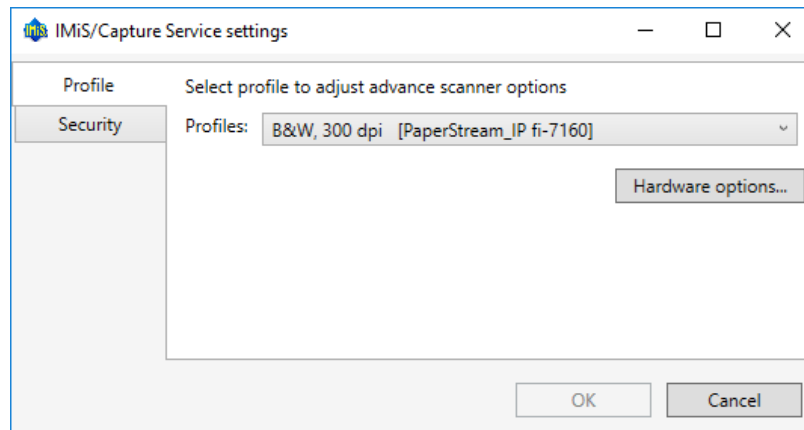
V kolikor v nastavitvenem oknu izbere gumb »Cancel«, se okno zapre brez shranjevanja izvedenih sprememb na profilu.

Opozorilo: Uporabnik z administracijskimi pooblastili mora po ponovnem zagonu IMiS®/CaptureService obvezno osvežiti IMiS®/wScan aplikacijo v brskalniku (MS Edge, Mozilla Firefox, Google Chrome, ...).

4.3.1 Dodatno nastavljanje profilov

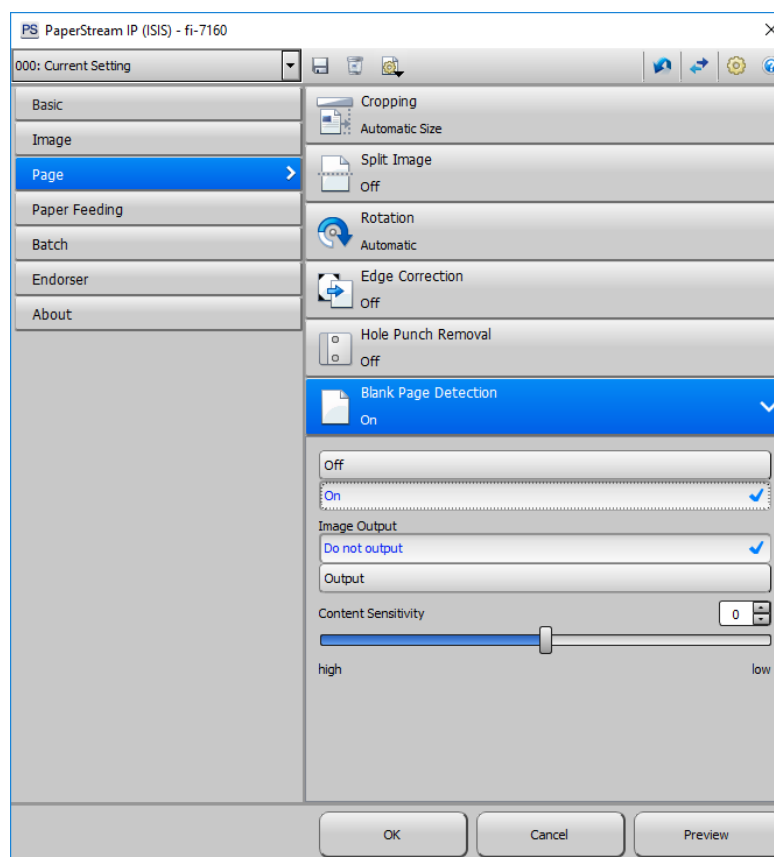
Za dodatne nastavitve profilov, uporabnik z administratorskimi pooblastili izbere zavihek »Profiles«.

Na desni strani se prikažejo nastavitve. V spustnem meniju »Profili« (angl. Profiles) so na voljo vsi profili, ki so določeni. Poleg imena profila je prikazan tudi model skenerja.



Slika 28: Pogovorno okno za nastavitve profilov in varnostne nastavitve

Uporabnik z administratorskimi pooblastili izbere profil za katerega želi izvesti dodatane skenerske nastavitve (npr. odstranjevanje praznih strani). Po izbiri profila klikne na gumb »Skenerske opcije...« (angl. Hardware options). Prikaže se nastavitveno okno skenerja.



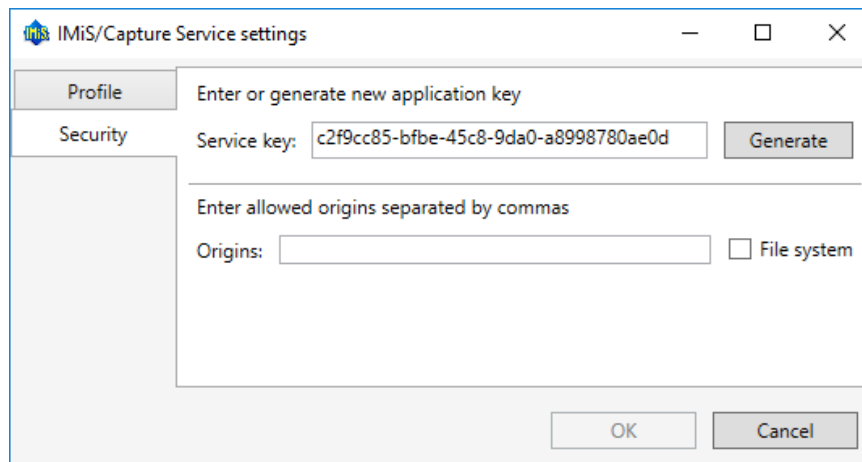
Slika 29: Nastavitveno okno Fujitsu PaperStream gonilnika

Opomba: Proizvajalci skenerjev imajo različno nastavitveno okno.

Spremembo nastavitve potrди s klikom na gumb »OK«.

4.3.2 Varnostne nastavitve

Uporabnik z administratorskimi pooblastili izbere zavihek »Varnost« (angl. Security). Prikažejo se varnostne nastavitve, ki vključujejo polje za vnos ali pridobitev varnostnega ključa ter vnos dovoljenih indirektnih spletnih domen.



Slika 30: Pogovorno okno za varnostne nastavitve

Uporabnik z administratorskimi pooblastili lahko vnese v polje »Varnostni ključ« (angl. Service key) poljuben niz znakov ali pa s klikom na gumb »Ustvari« (angl. Generate) pridobi nov enoličen niz znakov. Varnostni ključ razvijalec spletne aplikacije vnese v svojo aplikacijo tako, da ta niz znakov posreduje javascript knjižnici **imis.scan.js**.

Za dodatne informacije glej poglavje [2.4.2 Integracija knjižnice imis.scan.js](#).

Storitev IMiS®/Capture Service bo zavrnila vse REST zahteve s strani spletne aplikacije, ki ne bodo vsebovali enakega varnostnega ključa, kot je vnešen v polju »Varnostni ključ« (angl. Service key).

Uporabnik z administratorskimi pooblastili v polje »Izvori« (angl. Origins) vnese vse spletne domene, iz katerih lahko spletna aplikacija izvaja REST zahteve do storitve IMiS®/Capture Service. Posamezne vnešene spletne domene so med seboj ločene z vejico. Če je polje »Izvori« (angl. Origins) prazno, je dovoljen samo dostop preko lokalnega omrežnega vmesnika »localhost«. Za dostop iz vseh spletnih domen mora administrator vnesti znak »*« (zvezdica). V kolikor se spletna aplikacija izvaja neposredno iz datotečnega sistema mora uporabnik z administratorskimi pooblastili odključati polje »Datotečni sistem« (angl. File system).

***Opozorilo:** Uporabnik z administracijskimi pooblastili mora po ponovnem zagonu IMiS®/CaptureService obvezno osvežiti IMiS®/wScan aplikacijo v brskalniku (MS Edge, Mozilla Firefox, Google Chrome, ...).*

4.3.3 Administratorske dodatne nastavitve

Nastavljanje profilov je možno tudi mimo aplikacije IMiS®/wScan. To lahko izvede uporabnik z administratorskimi pravicami in z zanjem ažuriranja Windows registra.

Vse nastavitve profilov so zapisane v Windows registru pod ključem

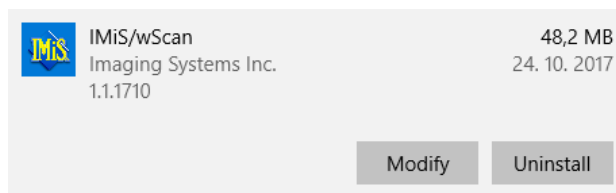
HKEY_LOCAL_MACHINE\SOFTWARE\Imaging Systems\IMiS Capture Service\ v polju profiles.

V kolikor storitev IMiS®/Capture Service nima pravic dostopa do tega ključa, se nastavitve shranijo na datotečni sistem v datoteko *profiles.json* v mapi *C:\ProgramData\Imaging Systems\IMiS Capture Service*.

Nastavitve profilov so zapisane v JSON datotečnem formatu. Zato jih je možno kopirati iz enega računalnika na drugega.

4.4 Odstranitev in spremembe

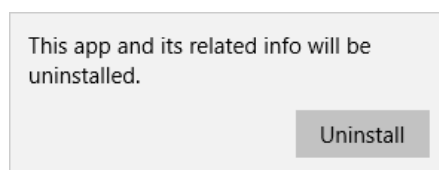
Sprememba v namestitvi ali odstranitev aplikacije IMiS®/wScan izvaja administrator na delovni postaji preko standardne Windows aplikacije »Add or Remove Programs«. Do aplikacije pride s klikom na gumb »Start«, poišče ikono »Nastavitve« (angl. Settings) in zažene »Add or Remove Programs«. Iz seznama aplikacij in funkcionalnih lastnosti (angl. Apps & features) izbere aplikacijo IMiS®/wScan.



Slika 31: Izbira med spremembo v namestitvi in odstranitvijo aplikacije

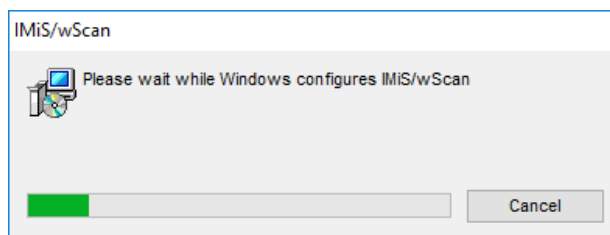
4.4.1 Odstranitev

Z izbiro možnosti »Odstrani« (angl. Uninstall) prične administrator s postopkom odstranitve aplikacije IMiS®/wScan.



Slika 32: Izbira odstranitve aplikacije

Med postopkom odstranitve se odstranijo vse datoteke in nastavitve aplikacije, ki jih je namestitveni paket ustvaril. Administrator lahko spremlja napredek pregleda konfiguracije preko pogovornega okna. S klikom na gumb »Cancel« prekliče postopek pregleda.

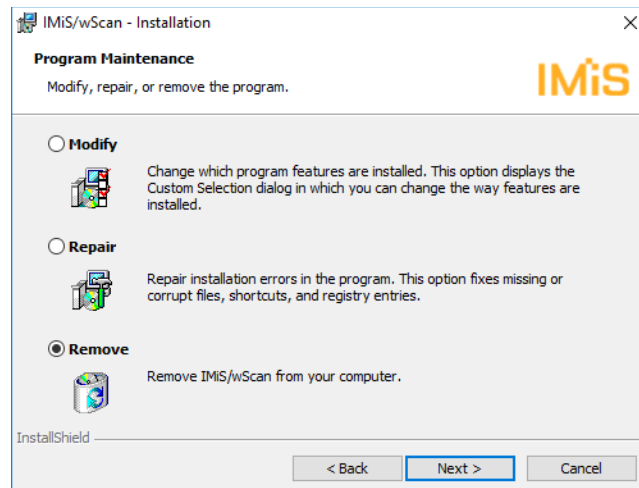


Slika 33: Prikaz vrstice napredka pregleda konfiguracije

V nadaljevanju postopka se administratorju prikaže pogovorno okno z možnostjo izbire:

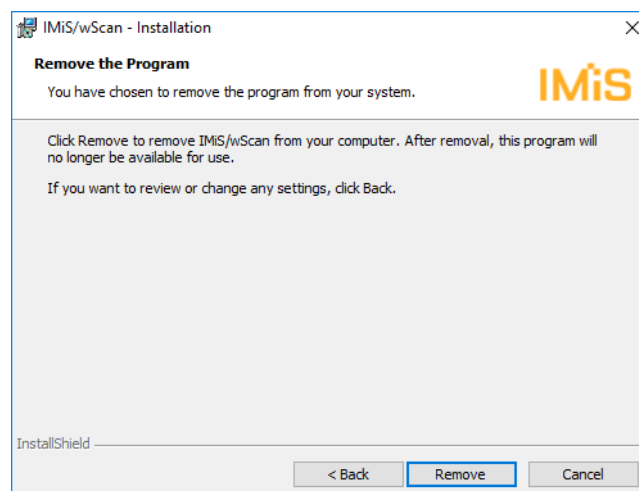
»Spremeni« (angl. Modify), »Popravi« (angl. Repair) ali »Odstrani« (angl. Remove).

Za odstranitev namestitvenega paketa izbere »Odstrani«. Izbiri potrdi z gumbom »Next«.



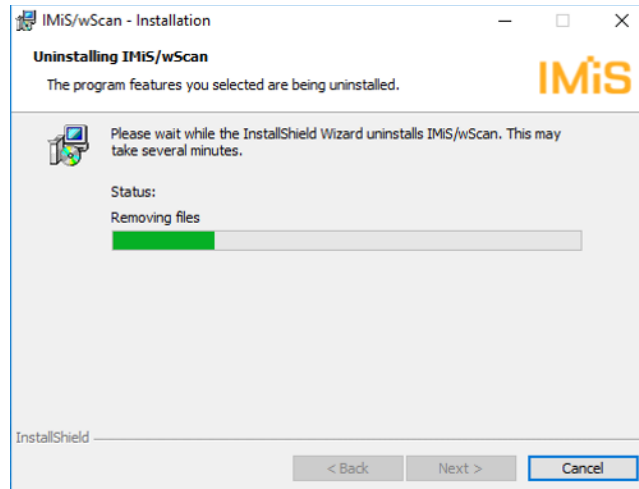
Slika 34: Izbira odstranitve aplikacije

V naslednjem koraku potrdi odstranitev s klikom na gumb »*Remove*«.



Slika 35: Potrditev odstranitve aplikacije

Odstranitev traja od nekaj sekund nekaj deset sekund, odvisno od različice namestitvenega paketa in hitrosti računalnika.



Slika 36: Prikaz vrstice napredka med postopkom odstranitve aplikacije

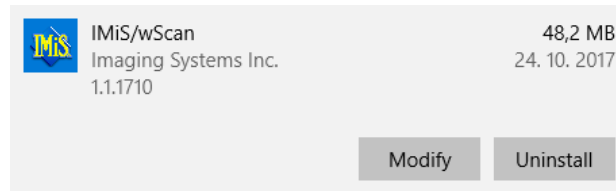
Po končani odstranitvi aplikacije se prikaže pogovorno okno, ki ga administrator zapre s klikom na gumb »*Finish*«.



Slika 37: Obvestilo o zaključku postopka odstranitve namestitvenega paketa

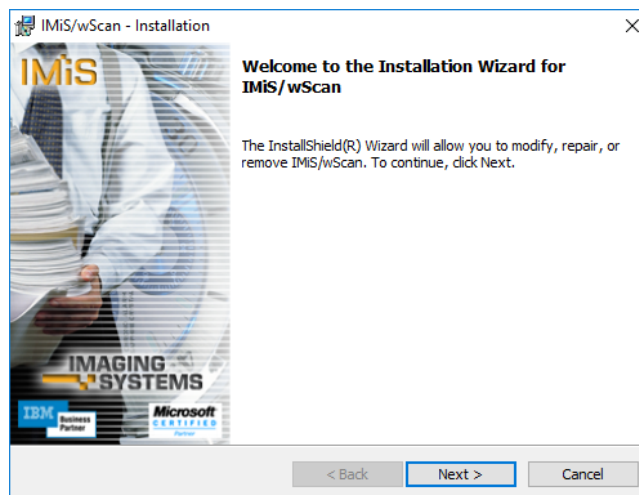
4.4.2 Spremembe in popravki v namestitvi

Administrator izvaja spremembe in popravke v namestitvi aplikacije IMiS®/wScan v Windows okolju preko gumba »Zaženi« (angl. Start), ikone »Nastavitve« (angl. Settings), »Add or Remove Programs« in izbrane aplikacije v seznamu aplikacij in funkcionalnih lastnosti (angl. Apps & features).



Slika 38: Izbira med spremembo in popravki v namestitvi ter odstranitvijo nameščene aplikacije

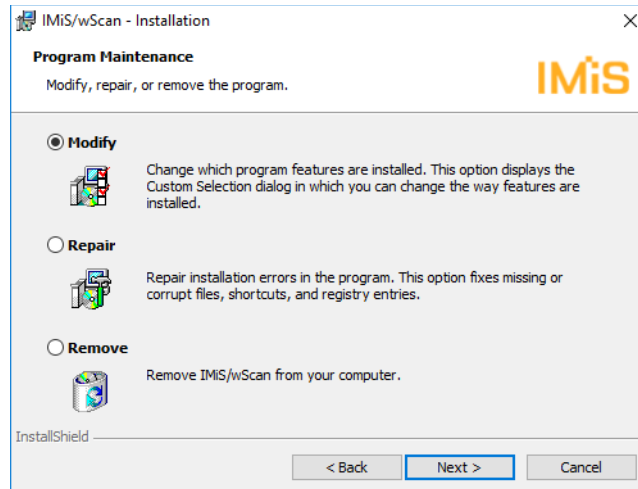
Administrator potrdi izbiro spremembe ali popravkov v namestitvi s klikom na gumb »Next«.



Slika 39: Zagon postopka izvedbe sprememb in popravkov v namestitvi

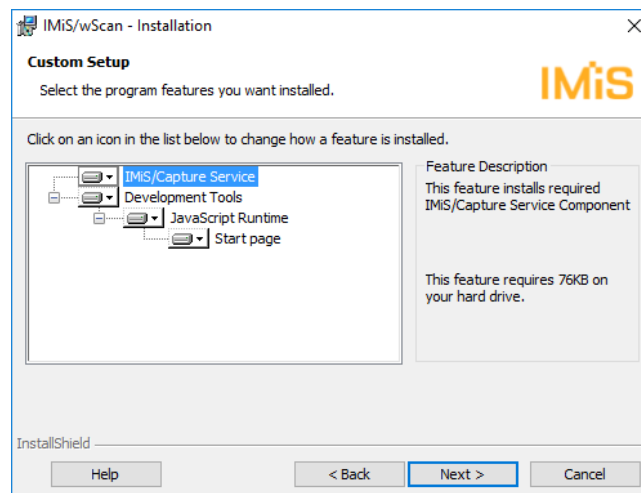
4.4.2.1 Spremembe v namestitvi

Administratorju se prikaže pogovorno okno z označeno možnostjo izbire »Spremeni« (angl. Modify). Izbiro potrdi z gumbom »Next«.



Slika 40: Izbira spremembe namestitve

Administrator s klikom na ikono označi elemente aplikacije, ki jih želi namestiti.



Slika 41: Izbira elementov pri spremembi namestitve

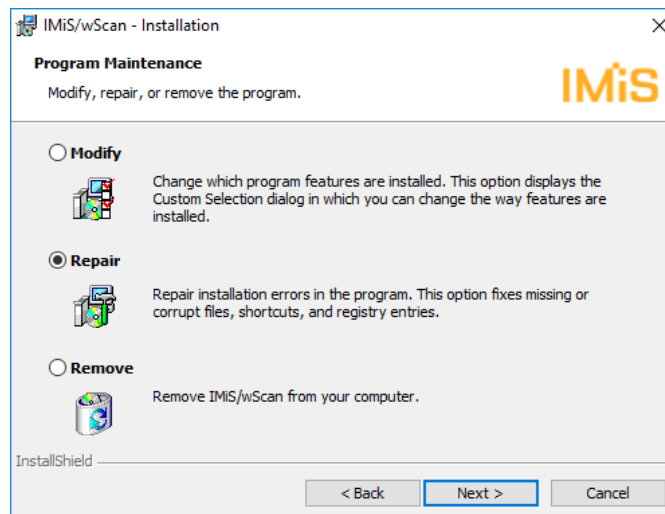
S potrditvijo izbire zažene postopek namestitve. Nadaljnji koraki so enaki kot pri običajni, polni in uporabniku prilagojeni namestitvi. Postopek se zaključi z namestitvijo vseh potrebnih elementov aplikacije. Za več informacij glej [poglavje 4.1.1 Namestitev s čarovnikom](#).

4.4.2.2 Popravki v namestitvi

V kolikor je med namestitvijo aplikacije IMiS®/wScan ali kasneje prišlo do poškodb namestitvenih datotek, bližnjic (angl. Shortcuts) ali vpisov v register (angl. Register entries), jih lahko administrator odpravi s popravki.

Administratorju se po zagonu postopka izvedbe sprememb in popravkov v namestitvi prikaže pogovorno okno z označeno možnostjo izbire »Popravi« (angl. Repair).

Izbiro potrdi z gumbom »Next«.



Slika 42: Izbira popravkov namestitve

V naslednjih nekaj korakih se izvedejo popravki namestitve. Postopek se zaključi z namestitvijo vseh potrebnih elementov aplikacije in ne zahteva posega administratorja..

Za več informacij glej [poglavje 4.1.1 Namestitev s čarovnikom](#).

4.5 Nadgradnja

Z izdajo nove verzije IMiS®/wScan je na vsaki posamezni delovni postaji potrebno izvesti postopek namestitve nove verzije . Postopek se izvede z namestitvenim »čarovnikom« in je enak kot pri namestitvi produkta.

Pri postopku nadgradnje se samodejno izvede odstranitev prejšnje različice produkta. Ohranijo se vse uporabniške nastavitve. Temu sledi postopek namestitve nove različice. Za več informacij glej [poglavje 4.1.1 Namestitve s čarovnikom](#).

5 TEHNIČNA DOKUMENTACIJA

Documentacija je pripravljena za razvijalce, ki imajo znanja Javascript programskega jezika ter poznajo koncepte objektno-usmerjenega programiranja. Razdeljena je na `imis.scan.js` in `imis.scan.ui.js` ter primere uporabe obeh knjižnic.

5.1 `imis.scan.js`

Javascript knjižnica **`imis.scan.js`** skrbi za izmenjavo podatkov s storitvijo IMiS®/Capture Service. Knjižnica je zgrajena na ECMAScript 6 standardu.

Knjižnica omogoča različne funkcionalnosti:

- dodajanje, branje, spreminjanje, brisanje profilov
- dodajanje, branje, izvajanje, zaustavitev opravil
- branje dokumentov
- branje strani na dokumentih
- branje barkod na strani
- branje modulov.

5.1.1 `imis.scan.Scan`

Objekt predstavlja metode za izmenjavo podatkov s strežnikom IMiS®/Capture Service. Omogoča branje, ustvarjanje, posodobitev in brisanje profilov, branje in ustvarjanje opravil. Omogoča zaznavo kreiranja in brisanja profilov ter kreiranja opravil. Pred pričetkom izvajanja knjižnice mora razvijalec pridobiti varnostni ključ za dostop do storitev IMiS®/Capture Service. Za pridobitev varnostnega ključa glej [poglavje 4.3.2 Varnostne nastavitve](#).

Konstruktor

imis.scan.Scan(options)	Ustvari nov objekt za izmenjavo podatkov in vzpostavi povezavo s strežnikom IMiS®/Capture Service.		
	Options objekt:		
	url	string	Naslov do IMiS®/Capture Service, privzeta vrednost je naslov gostitelja oz. računalnika (opcijsko).
	apiKey	string	Ključ za dostop do storitve IMiS®/Capture Service.
	onConnect	callback	Klic ob uspešni povezavi na IMiS®/Capture Service (opcijsko). callback: function()
	onConnectError	callback	Klic ob neuspešni povezavi na IMiS®/Capture Service (opcijsko). callback: function(error: string)
onError	callback	Klic ob napaki (opcijsko). callback: function(error: string)	

Metode

connect()	Vzpostavi povezavo s strežnikom IMiS®/Capture Service.
onCreateProfile(callback)	Klic ob kreiranju novega profila. Parametri: - callback: function(profile: imis.scan.Profile)
onDeleteProfile(callback)	Klic ob brisanju profila. Parametri: - callback: function(id: String)
onCreateJob(callback)	Klic ob kreiranju Joba. Parametri: - callback: function(job: imis.scan.Job)
onError(callback)	Klic ob napaki: Parametri: - callback: function(error: string)

getProfile(options)	<p>Vrne profil.</p> <p>Options objekt:</p> <table border="1" data-bbox="529 317 1375 541"> <tr> <td data-bbox="529 317 695 365">id</td> <td data-bbox="695 317 846 365">string</td> <td data-bbox="846 317 1375 365">Enolični identifikator profila</td> </tr> <tr> <td data-bbox="529 365 695 453">success</td> <td data-bbox="695 365 846 453">callback</td> <td data-bbox="846 365 1375 453">Klic ob uspešnem branju profila. callback: function(profile: imis.scan.Profile)</td> </tr> <tr> <td data-bbox="529 453 695 541">error</td> <td data-bbox="695 453 846 541">callback</td> <td data-bbox="846 453 1375 541">Klic ob neuspešnem branju profila. callback: function(error: string)</td> </tr> </table>			id	string	Enolični identifikator profila	success	callback	Klic ob uspešnem branju profila. callback: function(profile: imis.scan.Profile)	error	callback	Klic ob neuspešnem branju profila. callback: function(error: string)
id	string	Enolični identifikator profila										
success	callback	Klic ob uspešnem branju profila. callback: function(profile: imis.scan.Profile)										
error	callback	Klic ob neuspešnem branju profila. callback: function(error: string)										
getProfiles(options)	<p>Vrne zbirko profilov.</p> <p>Options objekt:</p> <table border="1" data-bbox="529 669 1375 877"> <tr> <td data-bbox="529 669 695 789">success</td> <td data-bbox="695 669 846 789">callback</td> <td data-bbox="846 669 1375 789">Klic ob uspešnem branju zbirke profilov. callback: function(profiles: imis.scan.Profile[])</td> </tr> <tr> <td data-bbox="529 789 695 877">error</td> <td data-bbox="695 789 846 877">callback</td> <td data-bbox="846 789 1375 877">Klic ob neuspešnem branju zbirke profilov. callback: function(error: string)</td> </tr> </table>			success	callback	Klic ob uspešnem branju zbirke profilov. callback: function(profiles: imis.scan.Profile[])	error	callback	Klic ob neuspešnem branju zbirke profilov. callback: function(error: string)			
success	callback	Klic ob uspešnem branju zbirke profilov. callback: function(profiles: imis.scan.Profile[])										
error	callback	Klic ob neuspešnem branju zbirke profilov. callback: function(error: string)										
createProfile(options)	<p>Ustvari profil.</p> <p>Options objekt:</p> <table border="1" data-bbox="529 1005 1391 1262"> <tr> <td data-bbox="529 1005 695 1054">profile</td> <td data-bbox="695 1005 902 1054">imis.scan.Profile</td> <td data-bbox="902 1005 1391 1054">Nov profil.</td> </tr> <tr> <td data-bbox="529 1054 695 1173">success</td> <td data-bbox="695 1054 846 1173">callback</td> <td data-bbox="846 1054 1391 1173">Klic ob uspešnem kreiranju profila. callback: function(profile: imis.scan.Profile)</td> </tr> <tr> <td data-bbox="529 1173 695 1262">error</td> <td data-bbox="695 1173 846 1262">callback</td> <td data-bbox="846 1173 1391 1262">Klic ob neuspešnem kreiranju profila. callback: function(error: string)</td> </tr> </table>			profile	imis.scan.Profile	Nov profil.	success	callback	Klic ob uspešnem kreiranju profila. callback: function(profile: imis.scan.Profile)	error	callback	Klic ob neuspešnem kreiranju profila. callback: function(error: string)
profile	imis.scan.Profile	Nov profil.										
success	callback	Klic ob uspešnem kreiranju profila. callback: function(profile: imis.scan.Profile)										
error	callback	Klic ob neuspešnem kreiranju profila. callback: function(error: string)										
updateProfile(options)	<p>Posodobi profil.</p> <p>Options objekt:</p> <table border="1" data-bbox="529 1390 1391 1642"> <tr> <td data-bbox="529 1390 695 1438">profile</td> <td data-bbox="695 1390 902 1438">imis.scan.Profile</td> <td data-bbox="902 1390 1391 1438">Profil.</td> </tr> <tr> <td data-bbox="529 1438 695 1558">success</td> <td data-bbox="695 1438 846 1558">callback</td> <td data-bbox="846 1438 1391 1558">Klic ob uspešni shranitvi profila. callback: function(profile: imis.scan.Profile)</td> </tr> <tr> <td data-bbox="529 1558 695 1642">error</td> <td data-bbox="695 1558 846 1642">callback</td> <td data-bbox="846 1558 1391 1642">Klic ob neuspešni shranitvi profila. callback: function(error: string)</td> </tr> </table>			profile	imis.scan.Profile	Profil.	success	callback	Klic ob uspešni shranitvi profila. callback: function(profile: imis.scan.Profile)	error	callback	Klic ob neuspešni shranitvi profila. callback: function(error: string)
profile	imis.scan.Profile	Profil.										
success	callback	Klic ob uspešni shranitvi profila. callback: function(profile: imis.scan.Profile)										
error	callback	Klic ob neuspešni shranitvi profila. callback: function(error: string)										

deleteProfile(options)	<p>Izbriše profil.</p> <p>Options objekt:</p> <table border="1" data-bbox="534 317 1398 541"> <tr> <td>profile</td> <td>imis.scan.Profile</td> <td>Profil.</td> </tr> <tr> <td>success</td> <td>callback</td> <td>Klic ob uspešnem izbrisu profila. callback: function()</td> </tr> <tr> <td>error</td> <td>callback</td> <td>Klic ob neuspešnem izbrisu profila. callback: function(error: string)</td> </tr> </table>		profile	imis.scan.Profile	Profil.	success	callback	Klic ob uspešnem izbrisu profila. callback: function()	error	callback	Klic ob neuspešnem izbrisu profila. callback: function(error: string)
profile	imis.scan.Profile	Profil.									
success	callback	Klic ob uspešnem izbrisu profila. callback: function()									
error	callback	Klic ob neuspešnem izbrisu profila. callback: function(error: string)									
getJob(options)	<p>Vrne opravilo.</p> <p>Options objekt:</p> <table border="1" data-bbox="534 669 1377 894"> <tr> <td>id</td> <td>string</td> <td>Enolični identifikator opravila.</td> </tr> <tr> <td>success</td> <td>callback</td> <td>Klic ob uspešnem branju opravila. callback: function(job: imis.scan.Job)</td> </tr> <tr> <td>error</td> <td>callback</td> <td>Klic ob neuspešnem branju opravila. callback: function(error: string)</td> </tr> </table>		id	string	Enolični identifikator opravila.	success	callback	Klic ob uspešnem branju opravila. callback: function(job: imis.scan.Job)	error	callback	Klic ob neuspešnem branju opravila. callback: function(error: string)
id	string	Enolični identifikator opravila.									
success	callback	Klic ob uspešnem branju opravila. callback: function(job: imis.scan.Job)									
error	callback	Klic ob neuspešnem branju opravila. callback: function(error: string)									
getJobs(options)	<p>Vrne opravila.</p> <p>Options objekt:</p> <table border="1" data-bbox="534 1022 1377 1199"> <tr> <td>success</td> <td>callback</td> <td>Klic ob uspešnem branju opravil. callback: function(job: imis.scan.Job[])</td> </tr> <tr> <td>error</td> <td>callback</td> <td>Klic ob neuspešnem branju opravil. callback: function(error: string)</td> </tr> </table>		success	callback	Klic ob uspešnem branju opravil. callback: function(job: imis.scan.Job[])	error	callback	Klic ob neuspešnem branju opravil. callback: function(error: string)			
success	callback	Klic ob uspešnem branju opravil. callback: function(job: imis.scan.Job[])									
error	callback	Klic ob neuspešnem branju opravil. callback: function(error: string)									
createJob(options)	<p>Ustvari opravilo, ob uspešnem kreiranju uniči zadnje ustvarjeno opravilo.</p> <p>Options objekt:</p> <table border="1" data-bbox="534 1327 1398 1583"> <tr> <td>profile</td> <td>string ali imis.scan.Profile</td> <td>Enolični identifikator profila ali profil.</td> </tr> <tr> <td>success</td> <td>callback</td> <td>Klic ob uspešnem kreiranju opravila. callback: function(job: imis.scan.Job)</td> </tr> <tr> <td>error</td> <td>callback</td> <td>Klic ob neuspešnem branju opravila. callback: function(error: string)</td> </tr> </table>		profile	string ali imis.scan.Profile	Enolični identifikator profila ali profil.	success	callback	Klic ob uspešnem kreiranju opravila. callback: function(job: imis.scan.Job)	error	callback	Klic ob neuspešnem branju opravila. callback: function(error: string)
profile	string ali imis.scan.Profile	Enolični identifikator profila ali profil.									
success	callback	Klic ob uspešnem kreiranju opravila. callback: function(job: imis.scan.Job)									
error	callback	Klic ob neuspešnem branju opravila. callback: function(error: string)									
getModules(options)	<p>Vrne zbirko modulov.</p> <p>Options objekt:</p> <table border="1" data-bbox="534 1711 1377 1881"> <tr> <td>success</td> <td>callback</td> <td>Klic ob uspešnem branju zbirke modulov. callback: function(job: imis.scan.Module[])</td> </tr> <tr> <td>error</td> <td>callback</td> <td>Klic ob neuspešnem branju zbirke modulov. callback: function(error: string)</td> </tr> </table>		success	callback	Klic ob uspešnem branju zbirke modulov. callback: function(job: imis.scan.Module[])	error	callback	Klic ob neuspešnem branju zbirke modulov. callback: function(error: string)			
success	callback	Klic ob uspešnem branju zbirke modulov. callback: function(job: imis.scan.Module[])									
error	callback	Klic ob neuspešnem branju zbirke modulov. callback: function(error: string)									

5.1.2 imis.scan.Profile

Objekt predstavlja profil. Omogoča zaznavo spremembe profila.

Konstruktor

imis.scan.Profile()	Ustvari nov profil.
---------------------	---------------------

Metode

setModule(module: imis.scan.Module)	Nastavi modul v profilu.
setModules(modules: imis.scan.Module [])	Nastavi zbirko modulov v profilu. Modul se odstrani v primeru, če ima nastavljeno lastnost remove na <i>true</i> , drugače se spremeni ali doda.
addModule(module: imis.scan.Module)	Doda nov modul v profil.
removeModule(module: imis.scan.Module)	Odstrani modul iz profila.
onChange(callback)	Klic ob spremembi profila. Parametri: - callback: function(profile: imis.scan.Profile)
clone()	Vrne kopijo.
equals(profile: imis.scan.Profile)	Vrne <i>true</i> če sta profila enaka, drugače <i>false</i> .

Lastnosti

id	string	Vrne enolični identifikator profila.
name	string	Vrne ali nastavi naziv profila.
disabled	boolean	Vrne ali je profil onemogočen.
disabledMessage	string	Vrne razlog za onemogočen profil.
source	string	Vrne ali nastavi identifikator modula na izvoru.
target	string	Vrne ali nastavi identifikator modula na ponoru.
scannerSource	imis.scan.ScannerModule	Vrne skenirni modul.
folderTarget	imis.scan.FolderTargetModule	Vrne končni modul, ki skrbi za shranjevanje.
barcodeExtractor	imis.scan.BarcodeExtractorModule	Vrne modul, ki razpozna barkode.
changed	boolean	Vrne ali je bil objekt spremenjen.
modules	imis.scan.Module []	Vrne zbirko modulov.

5.1.3 imis.scan.Job

Objekt predstavlja opravilo, omogoča zagon, preklic, zaznavo sprememb lastnosti in zaznavo kreiranja dokumenta.

Konstruktor

imis.scan.Job(options)	Ustvari nov objekt.
------------------------	---------------------

Metode

start(options)	Zažene opravilo.		
	Options objekt:		
	success	callback	Klic ob uspešnem zagonu opravila. callback: function(job: imis.scan.Job)
cancel(options)	Prekliče opravilo.		
	Options objekt:		
	success	callback	Klic ob uspešnem preklicu opravila. callback: function(job: imis.scan.Job)
onChange(callback)	Klic ob spremembi opravila.		
	Parametri:		
	- callback: function(job: imis.scan.Job)		
onCreateDocument(callback)	Klic ob kreiranju dokumenta.		
	Parametri:		
	- callback: function(document: imis.scan.Document)		

getJob(options)	<p>Vrne opravilo.</p> <p>Options objekt:</p> <table border="1" data-bbox="597 321 1395 573"> <tr> <td data-bbox="597 321 813 447">success</td> <td data-bbox="821 321 927 447">callback</td> <td data-bbox="935 321 1395 447">Klic ob uspešnem branju opravila. callback: function(job: imis.scan.Job)</td> </tr> <tr> <td data-bbox="597 457 813 573">error</td> <td data-bbox="821 457 927 573">callback</td> <td data-bbox="935 457 1395 573">Klic ob neuspešnem branju opravila. callback: function(error: string)</td> </tr> </table>	success	callback	Klic ob uspešnem branju opravila. callback: function(job: imis.scan.Job)	error	callback	Klic ob neuspešnem branju opravila. callback: function(error: string)
success	callback	Klic ob uspešnem branju opravila. callback: function(job: imis.scan.Job)					
error	callback	Klic ob neuspešnem branju opravila. callback: function(error: string)					
getDocuments(options)	<p>Vrne zbirko dokumentov.</p> <p>Options objekt:</p> <table border="1" data-bbox="597 709 1395 1052"> <tr> <td data-bbox="597 709 813 898">success</td> <td data-bbox="821 709 927 898">callback</td> <td data-bbox="935 709 1395 898">Klic ob uspešnem branju zbirke dokumentov. callback: function(documents: imis.scan.Document[])</td> </tr> <tr> <td data-bbox="597 909 813 1052">error</td> <td data-bbox="821 909 927 1052">callback</td> <td data-bbox="935 909 1395 1052">Klic ob neuspešnem branju zbirke dokumentov. callback: function(error: string)</td> </tr> </table>	success	callback	Klic ob uspešnem branju zbirke dokumentov. callback: function(documents: imis.scan.Document[])	error	callback	Klic ob neuspešnem branju zbirke dokumentov. callback: function(error: string)
success	callback	Klic ob uspešnem branju zbirke dokumentov. callback: function(documents: imis.scan.Document[])					
error	callback	Klic ob neuspešnem branju zbirke dokumentov. callback: function(error: string)					
getDocument(options)	<p>Vrne dokument.</p> <p>Options objekt:</p> <table border="1" data-bbox="597 1188 1395 1497"> <tr> <td data-bbox="597 1188 813 1335">success</td> <td data-bbox="821 1188 927 1335">callback</td> <td data-bbox="935 1188 1395 1335">Klic ob uspešnem branju dokumenta. callback: function(document: imis.scan.Document)</td> </tr> <tr> <td data-bbox="597 1346 813 1497">error</td> <td data-bbox="821 1346 927 1497">callback</td> <td data-bbox="935 1346 1395 1497">Klic ob neuspešnem branju dokumenta. callback: function(error: string)</td> </tr> </table>	success	callback	Klic ob uspešnem branju dokumenta. callback: function(document: imis.scan.Document)	error	callback	Klic ob neuspešnem branju dokumenta. callback: function(error: string)
success	callback	Klic ob uspešnem branju dokumenta. callback: function(document: imis.scan.Document)					
error	callback	Klic ob neuspešnem branju dokumenta. callback: function(error: string)					
getNextDocument(document: imis.scan.Document)	Vrne naslednji dokument (imis.scan.Document), če obstaja.						
getPrevDocument(document: imis.scan.Document)	Vrne prejšnji dokument (imis.scan.Document), če obstaja.						
onError(callback)	<p>Klic ob napaki na opravilu.</p> <p>Parametri:</p> <ul style="list-style-type: none"> - callback: function(error: string) 						

destroy()	Uniči opravilo.
download(callback, errorCallback)	Vrne URL do prenosa vseh dokumentov v opravilu. Parametri: - callback: function(uri: string) - errorCallback: function()
isCompleted()	Vrne ali je opravilo končano.
isCancelled()	Vrne ali je bilo opravilo preklicano.
isCreated()	Vrne ali je opravilo kreirano.
isInProgress()	Vrne ali se opravilo izvaja.
isPending()	Vrne ali je opravilo v vrsti za izvajanje.
isError()	Vrne ali se je zgodila napaka.

Lastnosti

id	string	Vrne enolični identifikator opravila.
index	number	Vrne indeks opravila.
name	string	Vrne naziv opravila.
status	number	Vrne status opravila. Nabor vrednosti: - -2: Opravilo je bilo preklicano - -1: Zgodila se je napaka - 0: Opravilo je končano - 1: Opravilo je kreirano - 2: Opravilo je v vrsti za izvajanje - 3: Opravilo se izvaja.
error	string	Vrne sporočilo o napaki na opravilu.
created	string	Vrne datum in čas nastanka opravila. Format 2017-10-02T09:58:15.9225533+02:00.
documentCount	number	Vrne število dokumentov v opravilu.
pageCount	number	Vrne število strani v vseh dokumentih v opravilu.
fileName	string	Vrne naziv datoteke opravila.

5.1.4 imis.scan.Document

Objekt predstavlja dokument, omogoča zaznavo ustvarjanja strani znotraj dokumenta, spremembo lastnosti dokumenta ter branje strani.

Metode

getFirstPage()	Vrne prvo stran imis.scan.Page v dokumentu.
getLastPage()	Vrne zadnjo stran imis.scan.Page v dokumentu.
getNextPage(page: imis.scan.Page)	Vrne naslednjo stran imis.scan.Page v dokumentu.
getPrevPage(page: imis.scan.Page)	Vrne prejšnjo stran imis.scan.Page v dokumentu.
onChange(callback)	Klic ob spremembi dokumenta. Vrne isti dokument s spremenjenimi lastnosti. Parametri: - callback: function(document: imis.scan.Document)
onCreatePage(callback)	Klic ob kreiranju strani znotraj dokumenta. Parametri: - callback: function(page: imis.scan.Page)
onError(callback)	Klic ob napaki na dokumentu. Parametri: - callback: function(error: string)
destroy()	Uniči dokument.
download(callback, errorCallback)	Pridobi povezavo za prenos dokumenta. Parametri: - callback: function(uri: string) - errorCallback: function()

Lastnosti

id	string	Vrne enolični identifikator dokumenta.
name	string	Vrne naziv dokumenta.
index	number	Vrne indeks dokumenta.
mime	string	Vrne tip vsebine dokumenta.
pageCount	number	Vrne število strani v dokumentu.

created	string	Vrne datum in čas nastanka. Format 2017-10-02T09:56:26.4618227+02:00
length	number	Vrne velikost dokumenta v bajtih.
fileName	string	Vrne naziv datoteke dokumenta.

5.1.5 imis.scan.Page

Objekt predstavlja stran, omogoča zaznavo sprememb lastnosti, branje predogleda strani in branje strani v osnovni velikosti.

Metode

getThumbnail(options)	<p>Pridobi povezavo do predogleda strani v formatu image/png.</p> <p>Options objekt:</p> <table border="1"> <tr> <td>height</td> <td>number</td> <td>Višina predogleda strani.</td> </tr> <tr> <td>width</td> <td>number</td> <td>Širina predogleda strani.</td> </tr> <tr> <td>success</td> <td>callback</td> <td>Klic ob uspešni pridobitvi povezave. Parametri: - callback: function(uri: string)</td> </tr> <tr> <td>error</td> <td>callback</td> <td>Klic ob neuspešni pridobitvi povezave. Parametri: - callback: function()</td> </tr> </table>		height	number	Višina predogleda strani.	width	number	Širina predogleda strani.	success	callback	Klic ob uspešni pridobitvi povezave. Parametri: - callback: function(uri: string)	error	callback	Klic ob neuspešni pridobitvi povezave. Parametri: - callback: function()
height	number	Višina predogleda strani.												
width	number	Širina predogleda strani.												
success	callback	Klic ob uspešni pridobitvi povezave. Parametri: - callback: function(uri: string)												
error	callback	Klic ob neuspešni pridobitvi povezave. Parametri: - callback: function()												
getImage(callback, errorCallback)	<p>Pridobi povezavo do strani v formatu image/png.</p> <p>Parametri:</p> <ul style="list-style-type: none"> - callback: function(uri: string) - errorCallback: function() 													
onChange(callback)	<p>Klic ob spremembi strani. Vrne stran s spremenjenimi lastnosti.</p> <p>Parametri:</p> <ul style="list-style-type: none"> - callback: function(page: imis.scan.Page) 													

Lastnosti

id	string	Vrne enolični identifikator strani.
index	number	Vrne indeks strani.
width	number	Vrne širino strani.

height	number	Vrne višino strani.
xresolution	number	Vrne horizontalno ločljivost strani v DPI.
yresolution	number	Vrne vertikalno ločljivost strani v DPI.
barcodes	imis.scan.Barcode[]	Vrne zbirko barkod na strani.
colorFormat	imis.scan.ColorFormat	Vrne format barve strani.

5.1.6 imis.scan.Barcode

Objekt predstavlja lastnosti razpoznane barkode na strani.

Lastnosti

height	number	Vrne višino barkode.
width	number	Vrne širino barkode.
text	string	Vrne razpoznano vsebino barkode.
point1	number	Vrne zgornjo levo točko barkode.
point2	number	Vrne zgornjo desno točko barkode.
point3	number	Vrne spodnjo desno točko barkode.
point4	number	Vrne spodnjo levo točko barkode.
posX	number	Vrne horizontalni odmik barkode na strani.
posY	number	Vrne vertikalni odmik barkode na strani.
type	string	Vrne vrsto barkode.

5.1.7 imis.scan.Module

Objekt predstavlja osnovo modula iz katerega so izpeljani različni moduli.

Metode

clone()	Vrne kopijo.
---------	--------------

Lastnosti

id	string	Vrne enolični identifikator modula.
sendTo	string[]	Vrne ali nastavi zbirko identifikatorjev modulov, katerim modul pošlje podatke.

type	string	Vrne vrsto modula. Nabor vrednosti: - scanner_source - barcode_extractor - folder_target
remove	boolean	Vrne ali nastavi vrednost, ki določa ali se modul odstrani, upošteva se samo pri klicu metode setModules na imis.scan.Profile .

5.1.8 imis.scan.ScannerModule

Objekt predstavlja skenirni modul, ki omogoča branje in spreminjanje nastavitev skenerja.

Ta razred je izpeljan iz [imis.scan.Module](#).

Lastnosti

driverName	string	Vrne ali nastavi naziv gonilnika.
scannerModel	string	Vrne model skenerja.
scannerValues	imis.scan.ScannerValue []	Vrne zbirko vseh priključenih skenerjev.
paperSize	string	Vrne ali nastavi velikost papirja.
paperSizes	string[]	Vrne zbirko velikosti papirja za izbrani gonilnik (driverName).
resolution	number	Vrne ali nastavi resolucijo skeniranja.
resolutions	number[]	Vrne zbirko resolucij skeniranja za izbrani gonilnik (driverName).
colorFormat	imis.scan.ColorFormat	Vrne ali nastavi barvo skeniranja.
colorFormats	imis.scan.ColorFormat []	Vrne zbirko barv skeniranja za izbrani gonilnik (driverName).
duplex	boolean	Vrne ali nastavi ali je obojestransko skeniranje omogočeno.

5.1.9 imis.scan.FolderTargetModule

Objekt predstavlja končni modul, ki omogoča nastavljanje lastnosti shranjevanja datotek na datotečni sistem, z možnostjo nastavitve direktorija, formata datotek, barve in kompresije.

Ta razred je izpeljan iz [imis.scan.Module](#).

Lastnosti

folder	string	Vrne ali nastavi pot do direktorija.
fileRoot	string	Vrne ali nastavi naziv datoteke.
fileFormat	string	Vrne ali nastavi format datoteke.
fileFormats	string[]	Vrne nabor formatov datoteke. Nabor vrednosti: - BMP - GIF - TIFF - JPEG - PCX - PDF/A - PNG.
colorFormat	imis.scan.ColorFormat	Vrne ali nastavi format barve.
colorFormats	imis.scan.ColorFormat[]	Vrne zbirko formatov barve, zbirka je povezana s formatom datoteke (fileFormat).
compression	string	Vrne ali nastavi kompresijo.
compressions	string[]	Vrne zbirko kompresij, zbirka je povezana s formatom barve (colorFormat) in formatom datoteke.

5.1.10 imis.scan.BarcodeExtractorModule

Objekt predstavlja modul, ki omogoča zaznavo barkod na posamezni strani.

Ta razred je izpeljan iz [imis.scan.Module](#).

Lastnosti

types	string[]	Vrne ali nastavi zbirko tipov barkod za prepoznavo.
typesValues	string[]	Vrne zbirko tipov barkod. Nabor vrednosti: <ul style="list-style-type: none"> - addon2 - addon5 - australianpost - aztec - bcdmatrix - codabar - code25_datalogic - code25_iata - code25_industrial - code25_interleaved - code25_invert - code25_matrix - code32 - code39 - code93 - datamatrix - ean13 - ean8 - intelligentmail - pdf417 - postnet - qrcode - royalpost - type128 - ucc128 - upc_a - upc_e.
orientation	string	Vrne ali nastavi možnost orientacij barkod.
orientationValues	string[]	Vrne zbirko orientacij barkod. Nabor vrednost: <ul style="list-style-type: none"> - horizontal: Zaznava horizontalnih barkod - vertical: Zaznava vertikalnih barkod - both: Zaznava horizontalnih ali vertikalnih barkod - horizontalverzicaldiagonal: Zaznava horizontalnih, vertikalnih ali 45° barkod

mode	string	Vrne ali nastavi način zaznave barkod.
modeValues	string[]	Vrne zbirko načinov zaznave barkod. Nabor vrednosti: - normal: Normalni način, hitrejši od enhanced. - enhanced: Izboljšan način, omogoča boljšo zaznavo, zaznava je počasnejša.

5.1.11 imis.scan.ScannerValue

Objekt predstavlja lastnosti skenerja.

Lastnosti

driverName	string	Vrne naziv gonilnika.
scannerModel	string	Vrne model skenerja.
colorFormats	imis.scan.ColorFormat []	Vrne zbirko barv.
paperSizes	string[]	Vrne zbirko velikosti papirja.
resolutions	number[]	Vrne zbirko resolucij.

5.1.12 imis.scan.ColorFormat

Objekt predstavlja lastnosti format barve.

Lastnosti

colorMode	string	Vrne tip barve slike. Nabor vrednosti: - blackwhite: Črno-bela slika, - grayscale: Sivinska slika, - color: Barvna slika.
colorDepth	number	Vrne barvno globino slike.
photometric	string	Vrne način branja podatkov slike.
compressions	string[]	Vrne zbirko kompresij, zbirka obstaja samo pri branju lastnosti colorFormats na imis.scan.FolderTargetModule .

5.2 imis.scan.ui.js

Knjižnica omogoča enostavno uporabo komponent, ki jih lahko uporabimo za prikaz skeniranja. Za prikaz komponent skrbi glavna komponenta `imis.scan.ui.Scan`, kjer določimo vse komponente, ki jih bomo uporabljali.

Za delovanje je potrebna `imis.scan.js` knjižnica.

5.2.1 imis.scan.ui.Scan

Objekt predstavlja glavno komponento, ki skrbi za prikaz različnih komponent. Ob ustvarjanju te komponente se komponente inicializirajo.

Konstruktor

<code>imis.scan.ui.Scan(options: UIScanOptions)</code>	Ustvari nov objekt in inicializirajo se komponente.
--	---

Metode

<code>show()</code>	Vzpostavi se povezava s strežnikom IMiS®/Capture Service in podatki se naložijo v komponente.
---------------------	---

Lastnosti

<code>job</code>	<code>imis.scan.Job</code>	Trenutno opravilo.
------------------	----------------------------	--------------------

5.2.1.1 UIScanOptions

Objekt predstavlja možnosti nastavitvev `imis.scan.ui.Scan`.

Lastnosti

<code>url</code>	<code>string</code>	Naslov do IMiS®/Capture Service, privzeta vrednost je naslov gostitelja oz. računalnika (opcijsko).
<code>apiKey</code>	<code>string</code>	Ključ za dostop do IMiS®/Capture Service.
<code>notifications</code>	<code>boolean</code>	Prikaz obvestil v brskalniku, privzeta vrednost <code>true</code> (opcijsko).
<code>thumbnails</code>	imis.scan.ui.Thumbnails	Prikaz dokumentov in strani (opcijsko).
<code>settings</code>	imis.scan.ui.Settings	Nastavitve vseh profilov (opcijsko).

imageView	imis.scan.ui.ImageView	Prikaz izbrane strani (opcijsko).
images	imis.scan.ui.ImageScroll	Prikaz vseh strani (opcijsko).
status	imis.scan.ui.Status	Prikaz statusa (opcijsko).
imageDetails	imis.scan.ui.ImageDetails	Podrobnosti izbrane strani (opcijsko).
progress	imis.scan.ui.Progress	Prikaz stanja opravila (opcijsko).
buttons	UIScanButtonsOptions	Nastavitve gumbov.
useLocalStorage	boolean	Določa uporabo shranjevanja nastavitvev (trenutno izbrani profil) v brskalnik, če nastavimo vrednost false se shranjene nastavitve izbrišejo, privzeta vrednost true (opcijsko).

5.2.1.2 UIScanButtonsOptions

Objekt predstavlja možnosti nastavitvev gumbov.

Lastnosti

scan	imis.scan.ui.Button	Gumb za začetek skeniranja. Privzeta vrednost besedila je Scan.
continue	imis.scan.ui.Button	Gumb za nadaljevanje skeniranja (opcijsko). Privzeta vrednost besedila je Continue.
cancel	imis.scan.ui.Button	Gumb za preklic skeniranja (opcijsko). Privzeta vrednost besedila je Cancel.
download	imis.scan.ui.Button	Gumb za prenos vseh dokumentov skeniranja (opcijsko). Privzeta vrednost besedila je Save.
color	imis.scan.ui.ColorDropdownButton	Seznam možnih barv, če je na voljo skener, sprememba se shrani samo začasno za vsako začeto opravilo (opcijsko).
profiles	imis.scan.ui.ProfilesButton	Zbirka profilov in možnost začasnega urejanja izbranega profila, sprememba se shrani samo za vsako začeto opravilo (opcijsko).

5.2.2 imis.scan.ui.Button

Objekt predstavlja komponento gumb, ki predstavlja osnovno komponento, s katero nadziramo začetek, nadaljevanje, preklic in prenos opravila.



Slika 43: Komponenta gumb

Konstruktor

imis.scan.ui.Button(options)	Ustvari nov gumb.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu kot atribut id.
	text	string	Besedilo gumba (opcijsko).
	tooltip	string	Vsebina pojavnega obvestila pod gumbom (opcijsko).
	darkMode	boolean	Temni način prikaza. Privzeta vrednost: <i>false</i> (opcijsko).
	color	string	Barva besedila gumba (opcijsko).
	backgroundCo lor	string	Barva ozadja gumba (opcijsko).
	fontSize	string	Velikost besedila gumba (opcijsko).
	width	string	Minimalna širina gumba (opcijsko).
height	string	Višina gumba (opcijsko).	
onClick	callback	Klic ob kliku na gumb (opcijsko). Parametri: - callback: function()	

Metode

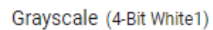
disable()	Onemogoči gumb.
enable()	Omogoči gumb.
showProgress()	Prikaže potek znotraj gumba.
hideProgress()	Skrije potek znotraj gumba.

Lastnosti

app	imis.scan.ui.Scan	Vrne glavno komponento.
-----	-------------------	-------------------------

5.2.3 imis.scan.ui.ColorDropDownButton

Objekt predstavlja komponento za izbiro barve skeniranja. Vrednost vpliva samo na opravilo, ki se bo začelo izvajati s pritiskom na gumb scan. Komponenta bo prikazala vrednosti, če obstaja skenirni modul na izbranem profilu in če ima vrednosti za izbro barve.



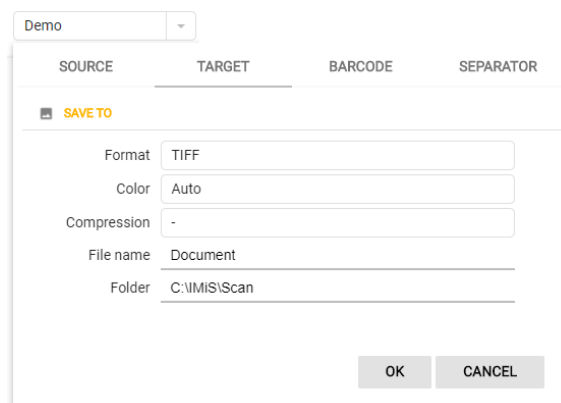
Slika 44: Komponenta za izbiro barve skeniranja

Konstruktor

imis.scan.ui.ColorDropDownButton(options)	Ustvari nov spustni meni za iziro barve skeniranja. Options objekt:		
id	string	Enolični identifikator elementa v HTML dokumentu kot atribut id.	

5.2.4 imis.scan.ui.ProfilesButton

Objekt predstavlja komponento za izbiro profila in spreminjanje nastavitev profila za skeniranje. Spremembe profila se bodo upoštevale samo pri vsakem začetku novega opravila. Izbran profil se bo shranil v brskalnik (Local Storage), če je shranjevanje omogočeno.



Slika 45: Komponenta za izbiro profila in spreminjanje nastavitev profila za skeniranje

Konstruktor

imis.scan.ui.ProfilesButton(options)	Ustvari nov spustni meni za izbiro profila in možnostjo spreminjanja nastavitve za novo opravilo.	
	Options objekt:	
	id	string Enolični identifikator elementa v HTML dokumentu kot atribut id.

5.2.5 imis.scan.ui.ImageDetails

Objekt predstavlja komponento za prikaz informacij trenutno izbrane strani.

Za delovanje je potrebna uporaba komponente [imis.scan.ui.ImageView](#) ali [imis.scan.ui.ImageScroll](#) v [imis.scan.ui.Scan](#).

Document	1 / 1
Name	Document_1
Created	16. 10. 2017 15:35:54
Type	image/tiff
Size	699.53 KB
Page	1 / 18
Width	2480
Height	3507
Resolution	300 dpi
Color	Black & White (1-Bit White0)
BARCODE	
Text	123456789012
Type	Type-128
Position	(935, 829)
BARCODE	
Text	ABCxyz#\$\$%15z
Type	Type-128
Position	(773, 2626)

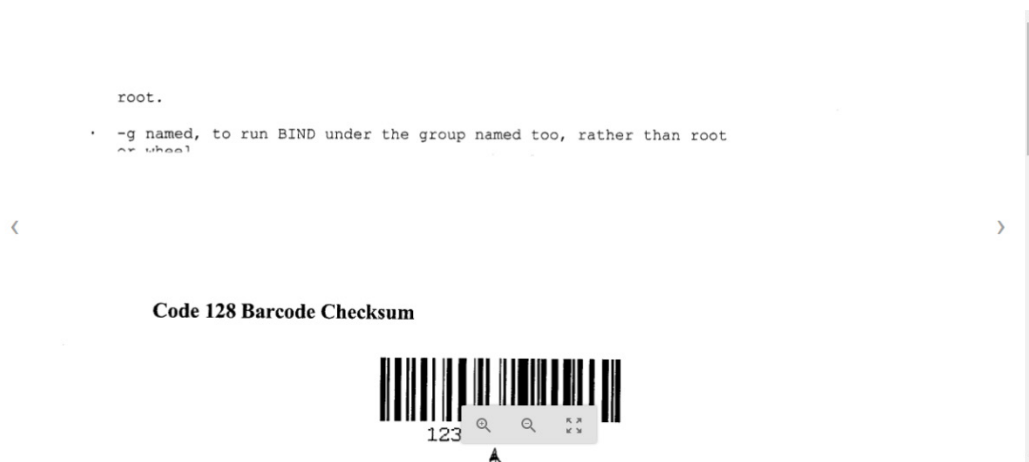
Slika 46: Komponenta za prikaz informacij trenutno izbrane strani

Konstruktor

imis.scan.ui.ImageDetails(option s)	Ustvari komponento za prikaz podrobnosti izbrane strani.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu kot atribut id.
	closed	boolean	Določa ali je komponenta skrita. Privzeta vrednost: <i>false</i> (opcijsko).
	close	boolean	Določa ali je gumb za zapiranje prikazan. Privzeta vrednost: <i>true</i> (opcijsko).
	darkMode	boolean	Temni način prikaza. Privzeta vrednost: <i>false</i> (opcijsko).
	onClose	callback	Klic ob kliku na gumb za zapiranje (opcijsko).
	background	string	Barva ozadja (opcijsko).
color	string	Barva besedila (opcijsko).	

5.2.6 imis.scan.ui.ImageView

Objekt predstavlja komponento za prikaz trenutno izbrane strani. Komponenta omogoča spreminjanje velikosti (povečevanje/pomanjšanje ali prikaz celotne strani), prehod na naslednjo ali prejšnjo stran.



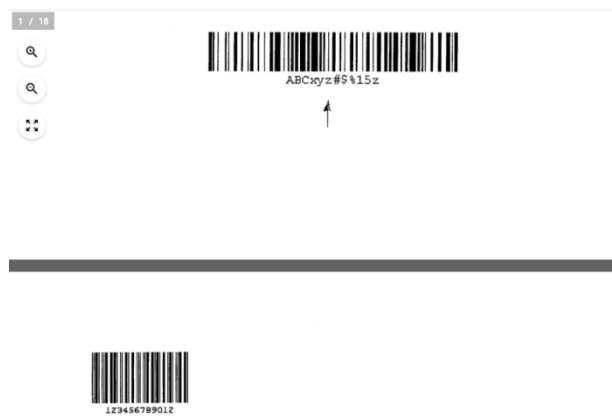
Slika 47: Komponenta za prikaz trenutno izbrane strani

Konstruktor

imis.scan.ui.ImageView(options)	Ustvari komponento za prikaz izbrane strani.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu kot atribut id.
	onPropertiesSelected	callback	Klic ob izbiri lastnosti v kontekstnem meniju (opcijsko). callback: function()
background	string	Barva ozadja (opcijsko).	

5.2.7 imis.scan.ui.ImageScroll

Objekt predstavlja komponento za prikaz zbirke strani. Komponenta omogoča spreminjanje velikosti (povečevanje/pomanjšanje ali prikaz celotne strani), prehod na naslednjo ali prejšnjo stran.



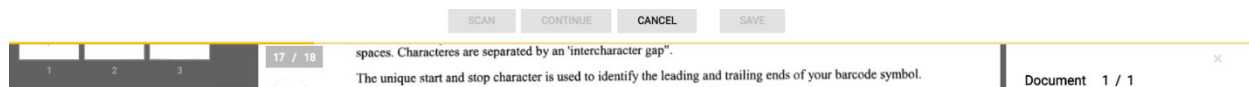
Slika 48: Komponenta za prikaz zbirke strani

Konstruktor

imis.scan.ui.ImageScroll(options)	Ustvari komponento za prikaz zbirke strani.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu kot atribut id.
	focusNewPage	boolean	Ob dodajanju nove strani se ta prikaže v vidnem polju. Privzeta vrednost: <i>true</i> (opcijsko).
	pageIndex	boolean	Prikaz pozicije trenutne strani. Privzeta vrednost: <i>true</i> (opcijsko).
	controls	boolean	Prikaz kontrol povečave, zmanjšanja in prilagajanja slike zaslona. Privzeta vrednost: <i>true</i> (opcijsko).
	darkMode	boolean	Temnejši način prikaza. Privzeta vrednost: <i>false</i> (opcijsko).
	contextMenu.enabled	boolean	Določa ali je omogočen kontekstni meni za posamezno stran (opcijsko).
	contextMenu.onPropertiesSelected	callback	Klic ob izbiri lastnosti v kontekstnem meniju (opcijsko). callback: function()
background	string	Barva ozadja (opcijsko).	

5.2.8 imis.scan.ui.Progress

Objekt predstavlja komponento za prikaz poteka trenutnega opravila.



Slika 49: Komponenta za prikaz trenutnega opravila

Konstruktor

imis.scan.ui.Progress(options)	Ustvari komponento za prikaz poteka trenutnega opravila.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu kot atribut id.
	darkMode	boolean	Temni način prikaza. Privzeta vrednost: <i>false</i> (opcijsko).
color	string	Barva komponente med izvajanjem opravila (opcijsko).	

5.2.9 imis.scan.ui.Status

Objekt predstavlja komponento za prikaz statusa, ki določa ali je povezava s strežnikom vzpostavljena.



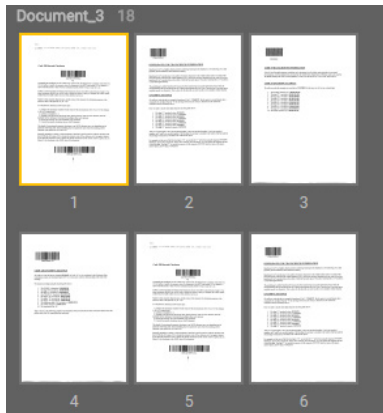
Slika 50: Komponenta za prikaz statusa

Konstruktor

imis.scan.ui.Status(options)	Ustvari komponento za prikaz statusa.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu.

5.2.10 imis.scan.ui.Thumbnails

Objekt predstavlja komponento za prikaz dokumentov in strani. Komponenta omogoča nastavljanje velikosti posameznih predogledov strani, orientacijo zbirke dokumentov ali način galerije, ki prikaže stran in podrobnosti v dialogu.



Slika 51: Komponenta za prikaz dokumentov

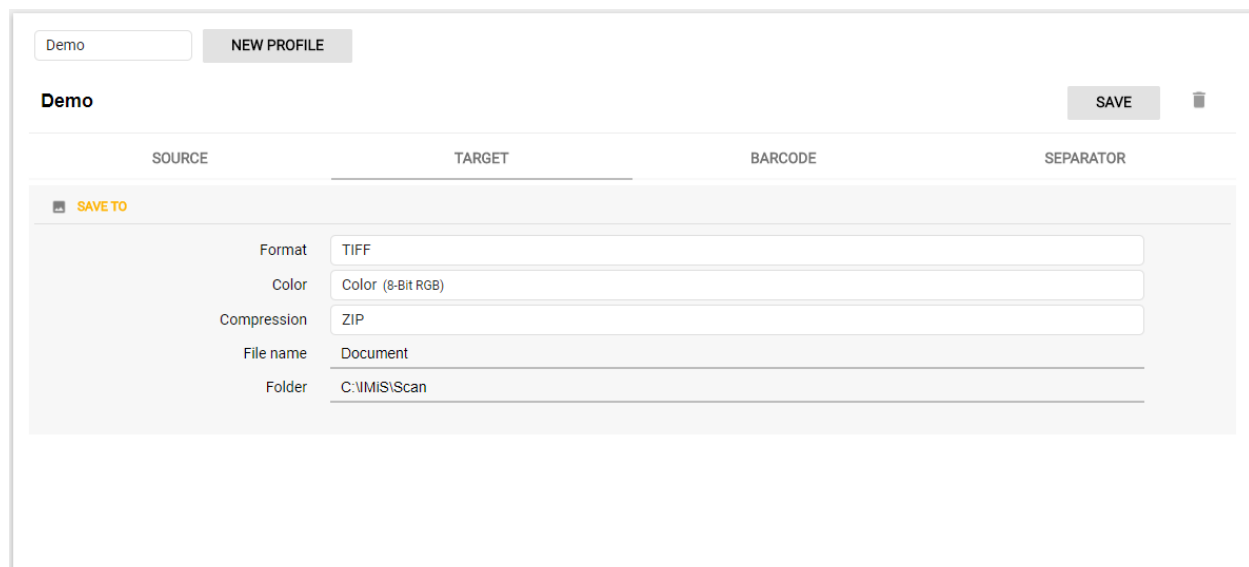
Konstruktor

imis.scan.ui.Thumbnails (options)	Ustvari komponento za prikaz strani kot predogled.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu.
	gallery	boolean	Način galerije, omogočen je dvoklik na posamezno stran, ki odpre dialog z povečano stranjo in podrobnostmi. Privzeta vrednost: <i>false</i> (opcijsko).
	focusNewPage	boolean	Ob dodajanju nove strani se ta prikaže v vidnem polju. Privzeta vrednost: <i>true</i> (opcijsko).
	darkMode	boolean	Temnejši način prikaza. Privzeta vrednost: <i>false</i> (opcijsko).
	orientation	string	Orientacija zbirke dokumentov. Nabor vrednosti: - horizontal, - vertical. Privzeta vrednost: horizontal (opcijsko).
	thumbnail.width	number	Širina strani (opcijsko).
	thumbnail.height	number	Višina strani (opcijsko). Privzeta vrednost: 150.
thumbnail.title	boolean	Določa ali je naslov strani prikazan. Privzeta vrednost: <i>true</i> (opcijsko).	
thumbnail.titleColor	string	Barva naslova strani (opcijsko).	

	contextMenu.enabled	boolean	Določa ali je omogočen kontekstni meni za posamezno stran (opcijsko).
	contextMenu.onPropertiesSelected	callback	Klic ob izbiri lastnosti v kontekstnem meniju (opcijsko). callback: function()
	color	string	Barva naslova dokumenta (opcijsko).
	backgroundColor	string	Barva ozadja (opcijsko).

5.2.11 imis.scan.ui.Settings

Objekt predstavlja komponento za nastavljanje profilov. Komponenta omogoča prikaz, dodajanje, spreminjanje ali brisanje profilov.



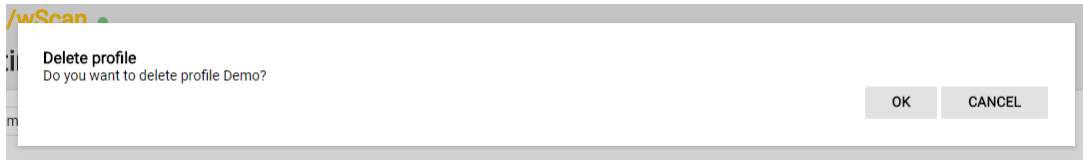
Slika 52: Komponenta za nastavljanje profilov

Konstruktor

imis.scan.ui.Settings(options)	Ustvari in prikaže novo komponento za prikaz nastavitev.		
	Options objekt:		
	id	string	Enolični identifikator elementa v HTML dokumentu.

5.2.12 imis.scan.ui.AlertDialog

Objekt predstavlja komponento za prikaz dialoga. Komponenta omogoča nastavljanje naslova, besedila in zaznavo ali je uporabnik potrdil ali preklical dialog.



Slika 53: Komponenta za prikaz dialoga

Konstruktor

imis.scan.ui.AlertDialog(options)	Ustvari in prikaže nov dialog.		
	Options objekt:		
	title	string	Naslov dialoga.
	text	string	Vsebina dialoga.
	ok	callback	Klic ob pritisku na gumb ok. callback: function()
cancel	callback	Klic ob pritisku na gumb cancel. callback: function()	

5.3 Primeri uporabe imis.scan.js

Primeri prikažejo uporabo knjižnice imis.scan.js. Pripravljeni primeri pomagajo razvijalcem, da se iz primerov lahko enostavno naučijo uporabe knjižnice. Prikazani so primeri branja profilov, spreminjanje profila, zagon opravila in brisanje opravila. Primeri so dostopni iz začetne strani, če so bili vključeni ob inštalaciji in jih je možno izvajati.

5.3.1 Branje profilov

Primer branja profilov prikaže osnovno uporabo knjižnice. Zbirka profilov bo ob uspešnem branju prikazana v elementu z identifikatorjem profiles, če se bo zgodila napaka bo vidna v elementu z identifikatorjem error.

Ob končanem nalaganju strani, se bo ustvaril scan objekt, s katerim bomo prebrali vse profile, če bodo uspešno prebrani jih bomo prikazali v seznamu, drugače pa se bo prikazala napaka.

```
<!DOCTYPE html>
<html>
<head>
  <title>imis.scan.js</title>
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="sample.css" />
</head>
<body class="sample">
  <h1>Sample</h1>
  <p>This example demonstrates reading scan profiles.</p>

  <div>Profiles:</div>
  <ol id="profiles"></ol>
  <div id="error"></div>

  <script src="../imis.scan.js"></script>
  <script>
    window.addEventListener('load', function () {
      try {
        // Profiles ordered list
        var ol = document.getElementById("profiles");

        // Create a scan object
        var scan = new imis.scan.Scan(
          apiKey = API_KEY);

        // Read profiles
        scan.getProfiles({
          success: function (profiles) {
            for (var i = 0; i < profiles.length; i++) {
              // Add profile to ordered list
              var li = document.createElement("li");
              li.innerHTML = profiles[i].name;
              ol.appendChild(li);
            }
          },
          error: function (error) {
            // Show error
            document.getElementById("error").innerHTML = error;
          }
        });
      } catch (e) {
        // Show error
        document.getElementById("error").innerHTML = e;
      }
    });
  </script>
</body>
</html>
```

5.3.2 Spreminjanje profila

Primer spreminjanja naziva profila, prikaže osnovo za spreminjanje lastnosti profila. Zbirka vseh profilov se naloži v element select z identifikatorjem profiles, kjer izberemo trenutni profil za spremembo. Klik na element button z identifikatorjem btn-update sproži posodobitev profila in posodobi zbirko profilov. Ob končanem nalaganju strani, se bo ustvaril scan objekt, s katerim bomo prebrali vse profile, če bodo uspešno prebrani jih bomo prikazali v seznamu in naložili naziv izbranega profila v vnosno polje, drugače pa se bo prikazala napaka. Ob spremembi profila v spustnem meniju se bo posodobilo vnosno polje z nazivom profila. Ob pritisku na gumb Update se bo nov naziv profila shranil v profil in sprememba se bo shranila na strežnik, ob uspešni shranitvi se bo posodobil seznam profilov, drugače pa se bo prikazala napaka.

```
<!DOCTYPE html>
<html>
<head>
  <title>imis.scan.js</title>
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="sample.css" />
</head>
<body class="sample">
  <h1>Sample</h1>
  <p>This example demonstrates updating profile name.</p>

  <h2>Update profile</h2>
  Select profile <select id="profiles"></select>

  <h3>Edit</h3>
  Profile name <input id="profile-name" type="text" placeholder="Profile name" />
  <button id="btn-update">Update</button>
  <div id="error"></div>

  <script src="../imis.scan.js"></script>
  <script>
    window.addEventListener('load', function () {
      var profilesList = [], // Profiles list
          selectedProfile = null, // Selected profile
          profilesSelectUI = document.getElementById("profiles"), // Profiles drop-
down list
          profileNameUI = document.getElementById("profile-name"); // Selected profile
name input text

      try {
        // Create a scan object
        var scan = new imis.scan.Scan(
          apiKey = API_KEY);

        // Load profiles to drop-down list
        var load = function () {
          scan.getProfiles({
            success: function (profiles) {
              profilesList = profiles;
              // Clear options
              profilesSelectUI.innerHTML = "";

```



```
    for (var i = 0; i < profiles.length; i++) {
        // Add profile option
        const profile = profiles[i];
        var option = document.createElement("option");
        option.value = profile.id;
        option.text = profile.name;
        if (null === selectedProfile)
            selectedProfile = profile;
        option.selected = profile.equals(selectedProfile)
        profilesSelectUI.add(option);
    }

    // Update selected profile name text input
    if (null !== selectedProfile)
        profileNameUI.value = selectedProfile.name;
},
error: function (error) {
    // Show error
    document.getElementById("error").innerHTML = error;
}
});
});
// Call load
load();

// Selected profile change listener
profilesSelectUI.addEventListener("change", function () {
    // Update selected profile
    selectedProfile = null;
    var selectedValue =
profilesSelectUI.options[profilesSelectUI.selectedIndex].value;
    for (var i = 0; i < profilesList.length; i++) {
        if (profilesList[i].id === selectedValue) {
            selectedProfile = profilesList[i];
            break;
        }
    }

    // Update selected profile name text input
    if (null !== selectedProfile)
        profileNameUI.value = selectedProfile.name;
});

// Update button click listener
document.getElementById("btn-update").addEventListener("click", function () {
    if (null == selectedProfile || null === profileNameUI.value || "" ===
profilesSelectUI.value ||
        selectedProfile.name === profileNameUI.value)
        return;

    // Update profile name
    selectedProfile.name = profileNameUI.value;

    // Save profile
    scan.updateProfile({
        profile: selectedProfile,
        success: function (profile) {
            // Load profiles
            load();
        },
        error: function (error) {
            // Show error

```

```

        document.getElementById("error").innerHTML = error;
    }
    });
});
} catch (e) {
    // Show error
    document.getElementById("error").innerHTML = e;
}
});
</script>
</body>
</html>

```

5.3.3 Zagon opravila

Primer zagona opravila, prikaže upravljanje opravila in prikaz rezultata skeniranja. Zbirka vseh profilov se naloži v element select z identifikatorjem profiles, kjer izberemo trenutni profil za kreiranje in zagon opravila. Klik na element button z identifikatorjem btn-start sproži kreiranje in zagon opravila. Status je prikazan v elementu z identifikatorjem job-progress. Prenos celotnega opravila je možen s klikom na elementu z identifikatorjem job-download, kadar je opravilo končalo.

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>imis.scan.js</title>
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="sample.css" />
</head>
<body class="sample">
  <h1>Sample</h1>
  <p>This example demonstrates starting job and displaying documents and pages.</p>

  <h2>Start Job</h2>
  <select id="profiles"></select>
  <button id="btn-start">Start</button>
  <br /><br />

  <div>Status: <span id="job-progress">None</span></div>
  <div id="error"></div>
  <a id="job-download" href="#">Download</a>
  <br /><br />

  <div><b>Documents</b></div>
  <div id="job">None</div>

  <script src="../imis.scan.js"></script>
  <script>
    window.addEventListener('load', function () {
      var profilesList = [], // Profiles list
          profilesSelectUI = document.getElementById("profiles"), // Profiles drop-
down list
          jobUI = document.getElementById("job"); // Job documents

      try {
        // Create a scan object
        var scan = new imis.scan.Scan(

```

```
    apiKey = API_KEY);

// Get profiles
scan.getProfiles({
  success: function (profiles) {
    profilesList = profiles;
    // Clear options
    profilesSelectUI.innerHTML = "";

    for (var i = 0; i < profiles.length; i++) {
      // Add profile option
      const profile = profiles[i];
      var option = document.createElement("option");
      option.value = profile.id;
      option.text = profile.name;
      profilesSelectUI.add(option);
    }
  },
  error: function (error) {
    console.error("getProfiles: " + error);
  }
});

// Start job button click listener
document.getElementById("btn-start").addEventListener("click", function () {
  // Clear download link
  document.getElementById("job-download").setAttribute("href", "#");

  // Find selected profile
  var selectedValue =
profilesSelectUI.options[profilesSelectUI.selectedIndex].value;
  var profile = null;
  for (var i = 0; i < profilesList.length; i++) {
    if (profilesList[i].id === selectedValue) {
      profile = profilesList[i];
      break;
    }
  }
  if (null == profile)
    return;

  // Create job
  scan.createJob({
    profile: profile.id,
    success: function (job) { // Job successfully created

      // Clear job documents
      jobUI.innerHTML = "";

      // Start job
      job.start({
        success: function () { // Job successfully started
          // Update job progress
          document.getElementById("job-progress").innerHTML =
imis.scan.JobStatus.toString[job.status];

          // Job changed callback
          job.onChange(function (job) {
            // Update job progress
            document.getElementById("job-progress").innerHTML =
imis.scan.JobStatus.toString[job.status];
            if (imis.scan.JobStatus.COMPLETED === job.status)
```

```
        document.getElementById("job-download").setAttribute("href",
job.download);
    });

    // Document created callback
    job.onCreateDocument(function (newDocument) {
        const documentElement = document.createElement("div");
        documentElement.style.marginBottom = "55px";

        // Document name
        const documentName = document.createElement("div");
        documentName.style.fontWeight = "bold";
        documentName.style.fontSize = "16px";
        documentName.innerHTML = newDocument.name + " [" +
newDocument.pageCount + "]";
        documentElement.appendChild(documentName);
        jobUI.appendChild(documentElement);

        // On create page callback
        newDocument.onCreatePage(function (page) {
            const pageElement = document.createElement("div");
            pageElement.style.display = "inline-block";

            // Page image
            const img = document.createElement("img");
            img.setAttribute("src", page.getThumbnailUri({ height: 150 }));
            pageElement.appendChild(img);

            // Page index
            const pageIndex = document.createElement("div");
            pageIndex.innerHTML = "Page " + page.index;
            pageElement.appendChild(pageIndex);

            documentElement.appendChild(pageElement);
        });

        // Document change callback
        newDocument.onChange(function (changedDocument) {
            documentName.innerHTML = changedDocument.name + " [" +
changedDocument.pageCount + "]";
        });
    },
    error: function (error) {
        // Show error
        document.getElementById("error").innerHTML = error;
    }
    });
    error: function (error) {
        // Show error
        document.getElementById("error").innerHTML = error;
    }
    });
} catch (e) {
    // Show error
    document.getElementById("error").innerHTML = e;
}
});
</script>
</body>
</html>
```

5.3.4 Brisanje profila

Primer brisanja profila prikaže osnovno uporabo knjižnice. Zbirka profilov bo ob uspešnem branju prikazana v spustnem meniju, kjer je možno izbirati profil za izbris. Profil bo izbrisan ob kliku na gumb delete in ob potrditvi dialoga o izbrisu. Primer izbriše profil, ki je shranjen na strežniku.

```
<!DOCTYPE html>
<html>
<head>
  <title>imis.scan.js</title>
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="sample.css" />
</head>
<body class="sample">
  <h1>Sample</h1>
  <p>This example demonstrates deleting profile.</p>

  <h2>Delete profile</h2>
  <select id="profiles"></select>
  <button id="btn-delete">Delete</button>
  <div id="error"></div>

  <script src="../imis.scan.js"></script>
  <script>
window.addEventListener('load', function () {
  var profilesList = [], // Profiles list
      profilesSelectUI = document.getElementById("profiles"); // Profiles drop-down
list

  try {
    // Create a scan object
    var scan = new imis.scan.Scan(
      apiKey = API_KEY);

    // Load profiles to drop-down list
    var load = function () {
      scan.getProfiles({
        success: function (profiles) {
          profilesList = profiles;
          // Clear options
          profilesSelectUI.innerHTML = "";

          for (var i = 0; i < profiles.length; i++) {
            // Add profile option
            const profile = profiles[i];
            var option = document.createElement("option");
            option.value = profile.id;
            option.text = profile.name;
            profilesSelectUI.add(option);
          }
        },
        error: function (error) {
          // Show error
          document.getElementById("error").innerHTML = error;
        }
      });
    };
  }
  // Call load
```

```
load();

document.getElementById("btn-delete").addEventListener("click", function () {
    // Find selected profile
    var selectedValue =
profilesSelectUI.options[profilesSelectUI.selectedIndex].value;
    var profile = null;
    for (var i = 0; i < profilesList.length; i++) {
        if (profilesList[i].id === selectedValue) {
            profile = profilesList[i];
            break;
        }
    }
    if (null == profile)
        return;

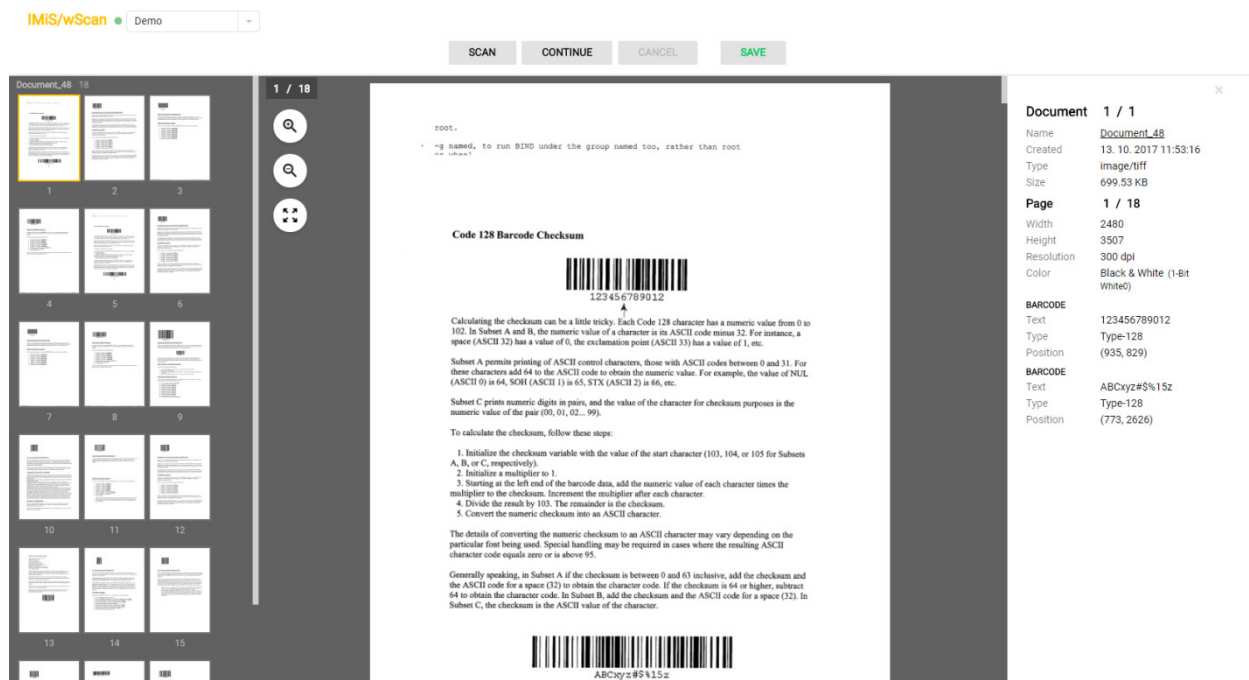
    // Show confirmation dialog
    if (confirm("Do you want do delete profile '" + profile.name + "'?")) {
        // Delete profile
        scan.deleteProfile({
            profile: profile,
            success: function () {
                // Load profiles
                load();
            },
            error: function (error) {
                // Show error
                document.getElementById("error").innerHTML = error;
            }
        });
    }
});
} catch (e) {
    // Show error
    document.getElementById("error").innerHTML = e;
}
});
</script>
</body>
</html>
```

5.4 Primeri uporabe imis.scan.ui.js

Primeri prikazujejo uporabo knjižnice »imis.scan.ui.js«. Pripravljeni primeri pomagajo razvijalcem, da se iz primerov lahko enostavno naučijo uporabe knjižnice. Prikazani so primeri klasičnega, modernega, klasičnega temnega načina in galerije. Primeri uporabljajo različne komponente, ki imajo določeno pozicijo na strani. Iz posameznih komponent lahko razvijalci sestavijo poljuben, aplikaciji prilagojen izgled strani. Primeri so dostopni iz začetne strani, če so bili vključeni ob namestitvi.

5.4.1 Klasičen način

Klasičen način (angl. Classic sample) je največkrat uporabljen primer uporabniškega vmesnika. Razvijalci aplikacij ga izberejo kadar želijo ohraniti tradicionalni izgled uporabniškega vmesnika. Pomanjšane slike strani dokumentov (angl. Thumbnails) se nahajajo v levem delu uporabniškega vmesnika. Razvrščene so glede na razpoložljiv prostor in velikost pomanjšanih slik (pokončno, ležeče). Strani dokumenta so prikazane v osrednjem delu. Uporabnik se premika med stranmi z drsnikom. Podrobnosti o straneh so razvrščene desno in jih lahko enostavno zapre ali pa ponovno odpre z izbiro menija na posamezni strani.



Slika 54: Primer uporabe klasičnega načina prikaza uporabniškega vmesnika

V primeru klasičnega načina so uporabljene naslednje komponente:

- imis.scan.ui.Thumbnails
- imis.scan.ui.ImageDetails
- imis.scan.ui.Status
- imis.scan.ui.Progress
- imis.scan.ui.Button
- imis.scan.ui.ProfilesButton

5.4.1.1 classic.html

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>IMiS/wScan Classic</title>
  <link rel="shortcut icon" type="image/png" href="img/favicon.png" />
  <link rel="stylesheet" href="imis.scan.ui.css" />
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="style/classic.css" />
</head>
<body>
  <div class="imis-scan-app">
    <nav id="nav-top">
      <div id="title" class="title"><a href="/">IMiS/wScan</a><div id="scan-
status"></div></div>
      <div id="imis-profile"></div>
    </nav>
    <nav id="nav" style="width: 500px; margin:auto;">
      <div id="scan-btn">Scan</div>
      <div id="continue-btn">Continue</div>
      <div id="cancel-btn">Cancel</div>
      <div id="download-btn">Save</div>
    </nav>
    <div id="imis-progress"></div>
    <div class="main" id="main">
      <div id="thumbnails" class="main-left">Thumbnails</div>
      <div id="images" class="main-center"></div>
      <div id="image-details" class="main-right">Details</div>
    </div>
  </div>
  <script src="imis.scan.js"></script>
  <script src="imis.scan.ui.js"></script>
</script>
window.addEventListener('load', function () {

  // Set scan version to title attribute
  document.getElementById("title").setAttribute("title", imis.scan.ui.version);

  // Create image details object
  const imageDetailsUI = new imis.scan.ui.ImageDetails({
    id: "image-details",
    onClose: function () {
      // Hide details
      imageDetailsUI.hide();
      // Resize images
      document.getElementById("images").setAttribute("style", "flex: 1");
    }
  });

  try {
    // Create a scan object
    var scan = new imis.scan.ui.Scan({
      //url: "http://example.com",
      apiKey = API_KEY,
      thumbnails: new imis.scan.ui.Thumbnails({
        id: "thumbnails",
        orientation: "vertical",
        thumbnail: {
          width: 80, // Thumbnail width
          title: true
        }
      })
    },

```



```

    contextMenu: {
        enabled: true,
        properties: true,
        onPropertiesSelected: function () {
            // Show details
            imageDetailsUI.show();
            // Reset images style
            document.getElementById("images").setAttribute("style", "");
        }
    }
}),
imageDetails: imageDetailsUI,
status: new imis.scan.ui.Status({ id: "scan-status" }),
progress: new imis.scan.ui.Progress({ id: "imis-progress" }),
images: new imis.scan.ui.ImageScroll({
    id: "images",
    controls: true,
    pageIndex: true,
    contextMenu: {
        onPropertiesSelected: function () {
            // Show details
            imageDetailsUI.show();
            // Reset images style
            document.getElementById("images").setAttribute("style", "");
        }
    }
}),
buttons: {
    scan: new imis.scan.ui.Button({ id: "scan-btn" }),
    profiles: new imis.scan.ui.ProfilesButton({id: "imis-profile" }),
    download: new imis.scan.ui.Button({ id: "download-btn" }),
    cancel: new imis.scan.ui.Button({ id: "cancel-btn" }),
    continue: new imis.scan.ui.Button({ id: "continue-btn" })
},
onError: function (message) {
    // Show dialog with error message
    new imis.scan.ui.AlertDialog({ title: "Error", text: message });
}
});
scan.show();
refreshLayoutHeight();
} catch (e) {
    console.error(e);
}
});

// Resizes height of main element
function refreshLayoutHeight() {
    var titleHeight = document.getElementById("nav-top").offsetHeight;
    var navHeight = document.getElementById("nav").offsetHeight +
document.getElementById("imis-progress").offsetHeight;
    var mainHeight = (window.innerHeight - titleHeight - navHeight);
    document.getElementById("main").style.height = mainHeight + "px";
}

// Resize event listener
window.addEventListener('resize', function (event) {
    refreshLayoutHeight();
});
</script>
</body>
</html>

```

5.4.1.2 classic.css

```
body {
  margin: 0;
  background: #fff;
  height: 100%;
  width: 100%;
  font-family: 'Roboto', sans-serif;
}

a {
  text-decoration: none;
  color: inherit;
}

.title {
  background: #fff;
  color: #FFC107;
  font-size: 20px;
  padding-top: 15px;
  font-weight: bold;
}

nav {
  margin: 0;
  position: relative;
  padding-bottom: 10px;
  padding-left: 25px;
  padding-right: 25px;
  background: #fff;
}

nav > div {
  display: inline-block;
  margin-right: 2px;
}

.main {
  background: #EEEEEE;
  display: flex;
  height: 500px;
  position: relative;
  box-shadow: 0 0 2px 2px rgba(0,0,0,0.075);
  z-index: 1000;
}

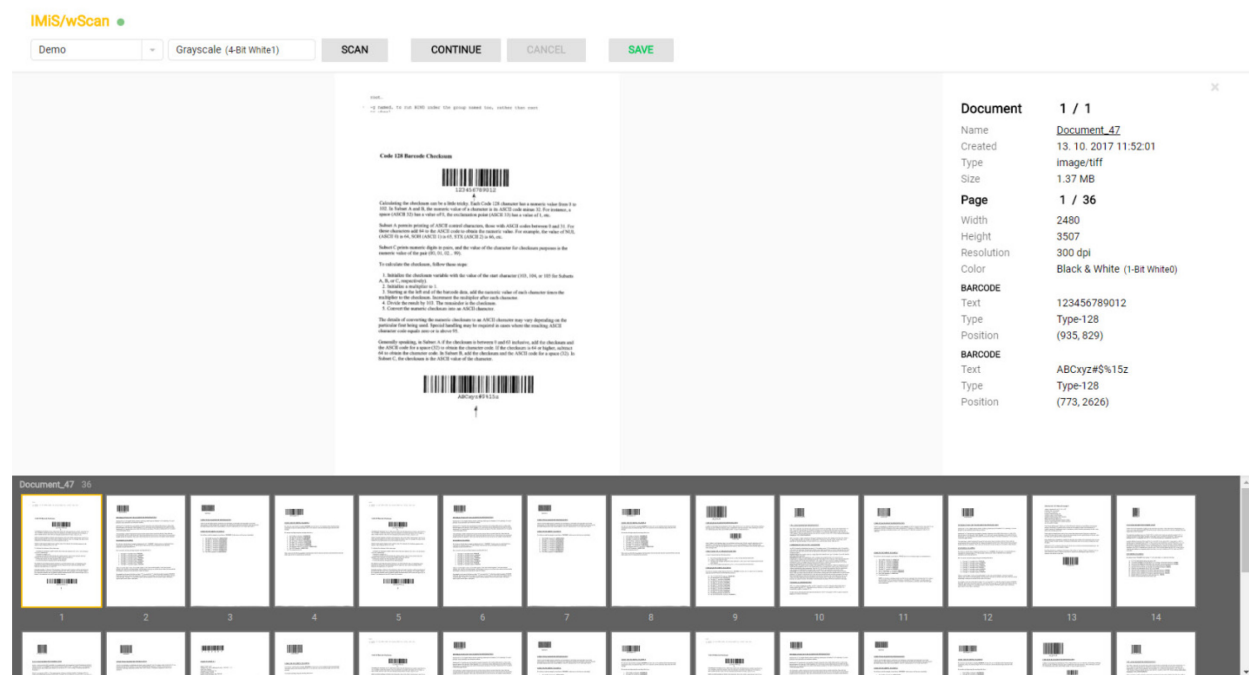
.main-left,
.main-right {
  flex: 0.25;
}

.main-center {
  flex: 0.50;
}

#download-btn {
  margin-left: 25px;
}
```

5.4.2 Moderni način

Moderni način (angl. Modern sample) je primeren za razvijalce aplikacij, ki sledijo novejšim smernicam izgleda uporabniškega vmesnika. Strani dokumenta zavzemajo pretežen del uporabniškega vmesnika. Podrobnosti o straneh se privzeto nahajajo na desni strani. Uporabnik jih lahko enostavno zapre ali pa ponovno odpre z izbiro menija na posamezni strani. Zaradi boljše preglednosti so pomanjšane slike strani dokumentov (angl. Thumbnails) razvrščene na dnu, kar je primerno zlasti v primeru večjih količin skeniranih dokumentov.



Slika 55: Primer uporabe modernega načina prikaza uporabniškega vmesnika

V primeru modernega načina so uporabljene naslednje komponente:

- imis.scan.ui.Thumbnails
- imis.scan.ui.ImageView
- imis.scan.ui.ImageDetails
- imis.scan.ui.Status
- imis.scan.ui.Progress
- imis.scan.ui.Button

- imis.scan.ui.ProfilesButton
- imis.scan.ui.ColorDropDownButton

5.4.2.1 modern.html

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>IMiS/wScan Modern</title>
  <link rel="shortcut icon" type="image/png" href="img/favicon.png" />
  <link rel="stylesheet" href="imis.scan.ui.css" />
  <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
  <link rel="stylesheet" href="style/modern.css" />
</head>
<body>
  <nav id="nav-top">
    <div id="title" class="title"><a href="/">IMiS/wScan</a><div id="scan-
status"></div></div>
  </nav>
  <nav id="nav">
    <div id="imis-profile"></div>
    <div id="imis-profile-color"></div>
    <div id="scan-btn"></div>
    <div id="continue-btn" style="margin-left:25px;"></div>
    <div id="cancel-btn"></div>
    <div id="download-btn"></div>
  </nav>
  <div id="imis-progress"></div>
  <div class="main" id="main">
    <div id="image-view" class="main-center"></div>
    <div id="image-details" class="main-right"></div>
  </div>
  <div id="thumbnails"></div>
  <script src="imis.scan.js"></script>
  <script src="imis.scan.ui.js"></script>
<script>
window.addEventListener('load', function () {

  //imis.scan.ui.Resource.lang = "si";

  document.getElementById("title").setAttribute("title", imis.scan.ui.version);

  try {
    var scan = new imis.scan.ui.Scan({
      //url: "http://beno:5000",
      //url: "https://beno:5443",
      //url: "http://localhost:5000",
      //notifications: false,
      //useLocalStorage: false,
      apiKey = API_KEY,
      thumbnails: new imis.scan.ui.Thumbnails({
        id: "thumbnails",
        //backgroundColor: "#f5f5f5",
        //color: "#0f0",
        //focusNewPage: false,
        //orientation: "horizontal",
        thumbnail: {
          //width: 25,
          //height: 250,

```

```

        //title: false
        //titleColor: "#00f"
    },
    contextMenu: {
        enabled: true,
        onPropertiesSelected: function() {
            document.getElementById("image-details").style.display = null;
            document.getElementById("image-view").style.width = null;
        }
    }
}),
imageView: new imis.scan.ui.ImageView({
    id: "image-view",
    onPropertiesSelected: function () {
        document.getElementById("image-details").style.display = null;
        document.getElementById("image-view").style.width = null;
    }
}),
imageDetails: new imis.scan.ui.ImageDetails({
    id: "image-details",
    //background: "#000",
    //color: "#0f0",
    onClose: function () {
        document.getElementById("image-details").style.display = "none";
        document.getElementById("image-view").style.width = "100%";
    }
}),
status: new imis.scan.ui.Status({ id: "scan-status" }),
progress: new imis.scan.ui.Progress({ id: "imis-progress" }),
buttons: {
    scan: new imis.scan.ui.Button({ id: "scan-btn" }),
    profiles: new imis.scan.ui.ProfilesButton({ id: "imis-profile" }),
    download: new imis.scan.ui.Button({ id: "download-btn" }),
    cancel: new imis.scan.ui.Button({ id: "cancel-btn" }),
    continue: new imis.scan.ui.Button({ id: "continue-btn" }),
    color: new imis.scan.ui.ColorDropDownButton({ id: "imis-profile-color" })
},
onError: function (message) {
    new imis.scan.ui.AlertDialog({ title: "Error", text: message });
}
});
scan.show();
refreshLayoutHeight();
} catch (e) {
    console.error(e);
    // Display error dialog
    new imis.scan.ui.AlertDialog({title: "Error", text: e});
}
});

// Resize main and thumbnails height
function refreshLayoutHeight() {
    var navHeight = document.getElementById("nav-top").offsetHeight +
        document.getElementById("nav").offsetHeight + document.getElementById("imis-
progress").offsetHeight;
    var thumbnailsHeight = window.innerHeight * 0.30; // 30% of window size
    // Main height
    document.getElementById("main").style.height = (window.innerHeight - navHeight -
thumbnailsHeight) + "px";
    // Thumbnails height
    document.getElementById("thumbnails").style.height = thumbnailsHeight + "px";
}

```

```
// Window resize event
window.addEventListener('resize', function (event) {
  console.log("resize");
  refreshLayoutHeight();
});
</script>
</body>
</html>
```

5.4.2.2 modern.css

```
body {
  margin: 0;
  background: #fff;
  height: 100%;
  width: 100%;
  font-family: 'Roboto', sans-serif;
}

a {
  text-decoration: none;
  color: inherit;
}

.title {
  background: #fff;
  color: #FFC107;
  font-size: 20px;
  padding-top: 15px;
  font-weight: bold;
}

nav {
  margin: 0;
  position: relative;
  padding-bottom: 10px;
  padding-left: 25px;
  padding-right: 25px;
  background: #fff;
}

nav > div {
  display: inline-block;
  margin-right: 2px;
}

.main {
  background: #EEEEEE;
  overflow: hidden;
  height: 500px;
  position: relative;
  box-shadow: 0 0 2px 2px rgba(0,0,0,0.075);
  z-index: 1000;
}

.main-left {
}

.main-center {
  float: left;
  width: 75%;
}
```

```

.main-right {
  float: left;
  width: 25%;
}

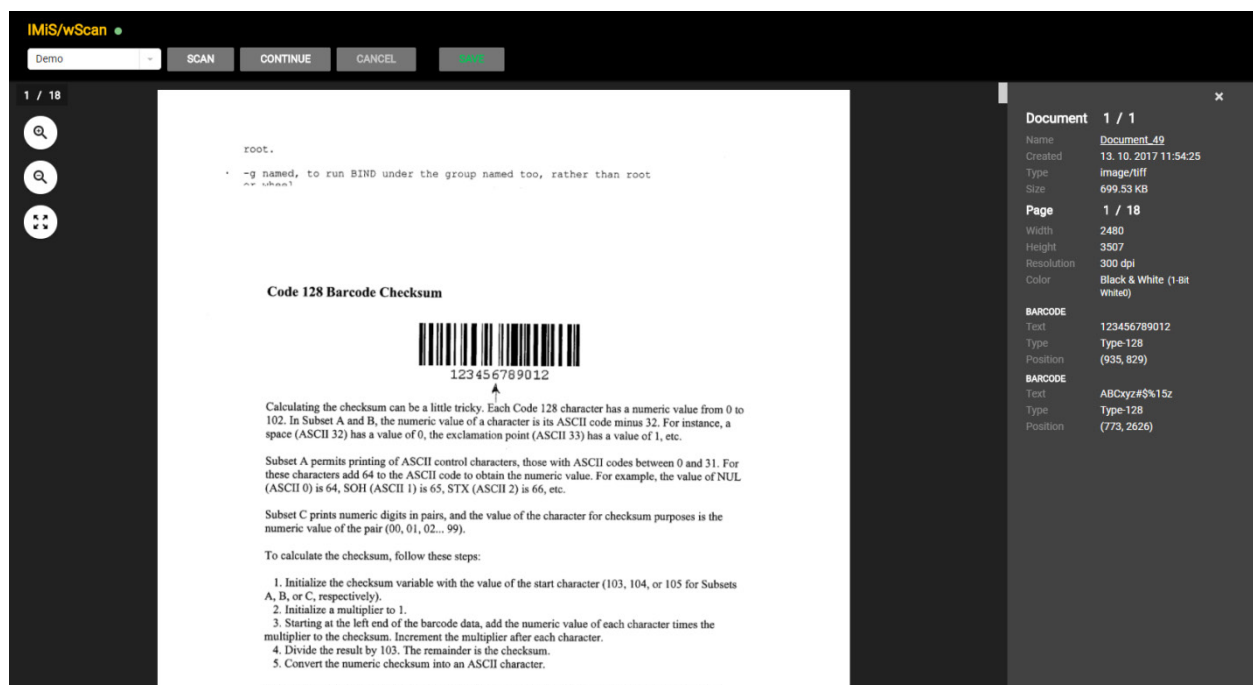
#download-btn {
  margin-left: 25px;
}

.imis-status {
  display: inline-block;
  margin-left: 10px;
}

```

5.4.3 Klasičen temni način

Klasičen temni način (angl. Classic dark sample) sledi zadnjim trendom pomembnejših pregledovalnikov dokumentov. Strani dokumenta so prikazane v osrednjem delu. Uporabnik se premika med stranmi z drsnikom. Dokumenti med seboj navidez niso ločeni. Iz podrobnosti o straneh je razvidno katerim dokumentom strani pripadajo. Podrobnosti o straneh se privzeto ne prikažejo. Z izbiro menija na posamezni strani jih uporabnik odpre. Prikažejo se v desnem delu uporabniškega vmesnika. Pomanjšane slike strani dokumentov (angl. Thumbnails) niso na voljo.



Slika 56: Primer uporabe klasičnega (temnega) načina prikaza uporabniškega vmesnika

V primeru klasičnega (temnega) načina so uporabljene naslednje komponente:

- imis.scan.ui.ImageScroll
- imis.scan.ui.ImageDetails
- imis.scan.ui.Status
- imis.scan.ui.Progress
- imis.scan.ui.Button
- imis.scan.ui.ProfilesButton

5.4.3.1 classic_dark.html

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>IMiS/wScan Classic (dark)</title>
  <link rel="shortcut icon" type="image/png" href="img/favicon.png" />
  <link rel="stylesheet" href="imis.scan.ui.css" />
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="style/classic.dark.css" />
</head>
<body>
  <nav id="nav-top">
    <div id="title" class="title"><a href="/">IMiS/wScan</a><div id="scan-
status"></div></div>
  </nav>
  <nav id="nav">
    <div id="imis-profile"></div>
    <div id="scan-btn">Scan</div>
    <div id="continue-btn">Continue</div>
    <div id="cancel-btn">Cancel</div>
    <div id="download-btn">Save</div>
  </nav>
  <div id="imis-progress"></div>
  <div class="main" id="main">
    <div id="images" class="main-center"></div>
    <div id="image-details" class="main-right">Details</div>
  </div>
  <script src="imis.scan.js"></script>
  <script src="imis.scan.ui.js"></script>
<script>
  window.addEventListener('load', function () {

    // Set scan version to title attribute
    document.getElementById("title").setAttribute("title", imis.scan.ui.version);

    // Create image details object
    var imageDetailsUI = new imis.scan.ui.ImageDetails({
      id: "image-details",
      closed: true,
      darkMode: true,
      onClose: function () {
        // Hide details
        imageDetailsUI.hide();
        // Resize images
        document.getElementById("images").style.flex = 1;
      }
    });
  });
</script>

```



```

    }
  });

  // Resize images
  document.getElementById("images").style.flex = 1;

  try {
    var scan = new imis.scan.ui.Scan({
      //url: "http://example.com",
      apiKey = API_KEY,
      imageDetails: imageDetailsUI,
      status: new imis.scan.ui.Status({ id: "scan-status" }),
      progress: new imis.scan.ui.Progress({ id: "imis-progress", darkMode: true }),
      images: new imis.scan.ui.ImageScroll({
        id: "images",
        //background: "#f5f5f5",
        darkMode: true,
        focusNewPage: false,
        contextMenu: {
          enabled: true,
          onPropertiesSelected: function () {
            imageDetailsUI.show();
            document.getElementById("images").style.flex = undefined;
          }
        }
      }),
      buttons: {
        scan: new imis.scan.ui.Button({ id: "scan-btn", darkMode: true }),
        profiles: new imis.scan.ui.ProfilesButton({ id: "imis-profile" }),
        download: new imis.scan.ui.Button({ id: "download-btn", darkMode: true }),
        cancel: new imis.scan.ui.Button({ id: "cancel-btn", darkMode: true }),
        continue: new imis.scan.ui.Button({ id: "continue-btn", darkMode: true })
      },
      onError: function (message) {
        // Show dialog with error message
        new imis.scan.ui.AlertDialog({ title: "Error", text: message });
      }
    });
    scan.show();
    refreshLayoutHeight();
  } catch (e) {
    console.error(e);
  }
});

// Resizes height of main element
function refreshLayoutHeight() {
  var titleHeight = document.getElementById("nav-top").offsetHeight;
  var navHeight = document.getElementById("nav").offsetHeight +
document.getElementById("imis-progress").offsetHeight;
  var mainHeight = (window.innerHeight - titleHeight - navHeight);
  document.getElementById("main").style.height = mainHeight + "px";
}

// Resize event listener
window.addEventListener('resize', function (event) {
  console.log("resize");
  refreshLayoutHeight();
});
</script>
</body>
</html>

```

5.4.3.2 classic.dark.css

```
body {
  margin: 0;
  background: #fff;
  height: 100%;
  width: 100%;
  font-family: 'Roboto', sans-serif;
}

a {
  text-decoration: none;
  color: inherit;
}

.title {
  color: #FFC107;
  font-size: 20px;
  padding-top: 15px;
  font-weight: bold;
}

nav {
  margin: 0;
  position: relative;
  padding-bottom: 10px;
  padding-left: 25px;
  padding-right: 25px;
  background: #000;
}

nav > div {
  display: inline-block;
  margin-right: 2px;
}

.main {
  background: #EEEEEE;
  display: flex;
  height: 500px;
  position: relative;
  box-shadow: 0 0 2px 2px rgba(255,255,255,0.1);
  z-index: 1000;
}

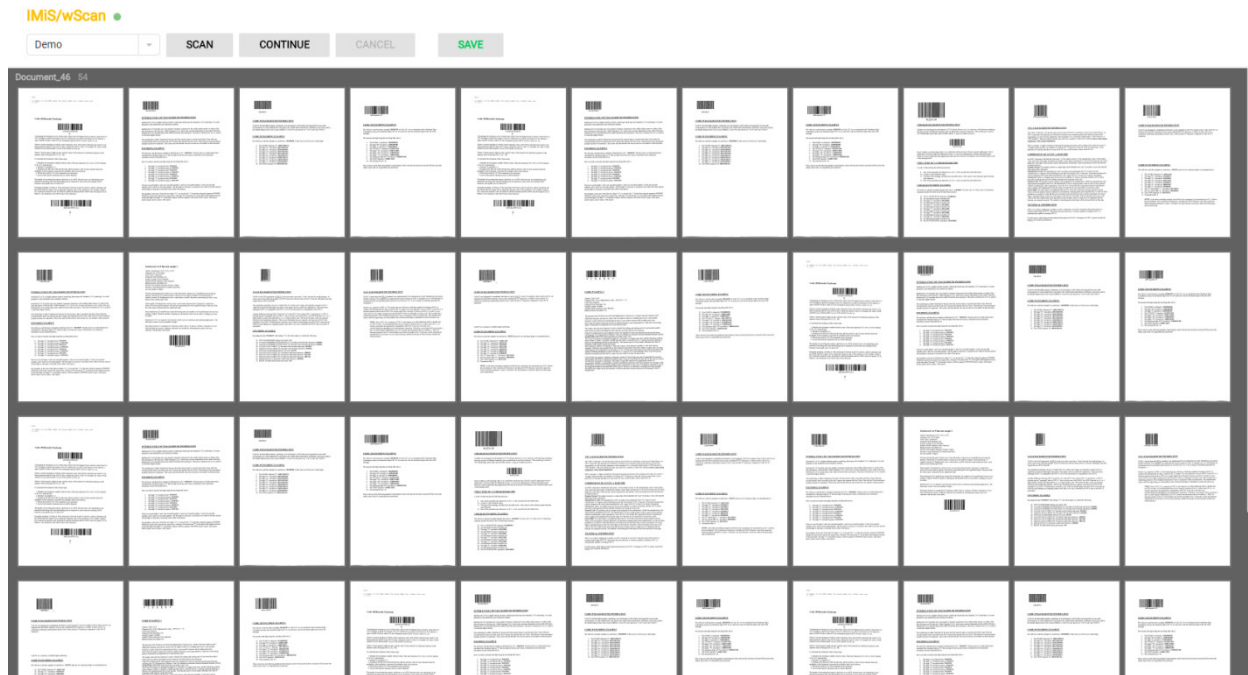
.main-center {
  flex: 0.75;
}

.main-right {
  flex: 0.25;
}

#download-btn {
  margin-left: 25px;
}
```

5.4.4 Galerija

Način Galerija (angl. Gallery sample) je primeren za prikaz večjih količin skeniranih dokumentov, oziroma za paketno skeniranje (angl. batch). Pomanjšane slike strani dokumentov (angl. Thumbnails) so večje in bolj vidne kot pri ostalih primerih. Prikazane so v vrsticah na celotnem uporabniškem vmesniku. Dokumenti so ločeni glede na število strani določenimi v nastavitvah. Podrobnosti o straneh se uporabniku prikažejo z dvoklikom na posamezni strani.



Slika 57: Primer uporabe galerija načina prikaza uporabniškega vmesnika

V primeru načina galerija so uporabljene naslednje komponente:

- imis.scan.ui.Thumbnails
- imis.scan.ui.Status
- imis.scan.ui.Progress
- imis.scan.ui.Button
- imis.scan.ui.ProfilesButton

5.4.4.1 gallery.html

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>IMiS/wScan Gallery</title>
  <link rel="shortcut icon" type="image/png" href="img/favicon.png" />
  <link rel="stylesheet" href="imis.scan.ui.css" />
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto" />
  <link rel="stylesheet" href="style/gallery.css" />
</head>
<body>
  <nav id="nav-top">
    <div id="title" class="title"><a href="/">IMiS/wScan</a><div id="scan-
status"></div></div>
  </nav>
  <nav id="nav">
    <div id="imis-profile"></div>
    <div id="scan-btn">Scan</div>
    <div id="continue-btn">Continue</div>
    <div id="cancel-btn">Cancel</div>
    <div id="download-btn">Save</div>
  </nav>
  <div id="imis-progress"></div>
  <div class="main" id="main">
    <div id="thumbnails">Thumbnails</div>
  </div>
  <script src="imis.scan.js"></script>
  <script src="imis.scan.ui.js"></script>
<script>
  window.addEventListener('load', function () {

    // Set scan version to title attribute
    document.getElementById("title").setAttribute("title", imis.scan.ui.version);

    try {
      const scan = new imis.scan.ui.Scan({
        //url: "http://example.com",
        apiKey = API_KEY,
        thumbnails: new imis.scan.ui.Thumbnails({
          id: "thumbnails",
          //darkMode: false,
          orientation: "horizontal",
          thumbnail: {
            height: 200, // thumbnail height
            title: false
          },
        },
        gallery: true,
        contextMenu: {
          enabled: false
        }
      }),
      status: new imis.scan.ui.Status({ id: "scan-status" }),
      progress: new imis.scan.ui.Progress({ id: "imis-progress" }),
      buttons: {
        scan: new imis.scan.ui.Button({ id: "scan-btn" }),
        profiles: new imis.scan.ui.ProfilesButton({ id: "imis-profile" }),
        download: new imis.scan.ui.Button({ id: "download-btn" }),
        cancel: new imis.scan.ui.Button({ id: "cancel-btn" }),
        continue: new imis.scan.ui.Button({ id: "continue-btn" })
      },
      onError: function (message) {

```

```

        // Show dialog with error message
        new imis.scan.ui.AlertDialog({ title: "Error", text: message });
    }
});
scan.show();
refreshLayoutHeight();
} catch (e) {
    console.error(e);
    // Show dialog with error message
    new imis.scan.ui.AlertDialog({ title: "Error", text: e });
}
});

// Resizes height of main element
function refreshLayoutHeight() {
    var titleHeight = document.getElementById("nav-top").offsetHeight;
    var navHeight = document.getElementById("nav").offsetHeight +
document.getElementById("imis-progress").offsetHeight;
    document.getElementById("main").style.height = (window.innerHeight - titleHeight -
navHeight) + "px";
}

// Resize event listener
window.addEventListener('resize', function (event) {
    refreshLayoutHeight();
});
</script>
</body>
</html>

```

5.4.4.2 gallery.css

```

body {
    margin: 0;
    background: #fff;
    height: 100%;
    width: 100%;
    font-family: 'Roboto', sans-serif;
}

a {
    text-decoration: none;
    color: inherit;
}

.title {
    background: #fff;
    color: #FFC107;
    font-size: 20px;
    padding-top: 15px;
    font-weight: bold;
}

nav {
    margin: 0;
    position: relative;
    padding-bottom: 10px;
    padding-left: 25px;
    padding-right: 25px;
    background: #fff;
}

```

```
nav > div {
  display: inline-block;
  margin-right: 2px;
}

.main {
  position: relative;
  z-index: 1000;
}

#download-btn {
  margin-left: 25px;
}
```

6 ODPRAVLJANJE TEŽAV

6.1 Težave pri uporabi IMiS®/wScan

V nadaljevanju so navedene pogoste težave pri uporabi aplikacije IMiS®/wScan, opis težav in navodilo za odpravo težav.

6.1.1 Napaka »Forbidden«

Vzrok težave

Spletna aplikacija preko knjižnice »imis.scan.js« ni posredovala pravih varnostnih ključev storitvi IMiS®/Capture Service.

Odprava težave

Uporabnik z administratorskimi pooblastili mora preveriti, da je varnostni ključ enak tako v storitvi IMiS®/Capture Service, kakor tudi v spletni aplikaciji. V kolikor je v storitvi IMiS®/Capture Service vnešen oziroma ustvarjen nov varnostni ključ, mora uporabnik z administratorskimi pooblastili izvesti ponovni zagon storitve. Po zagonu mora osvežiti spletno stran brskalnika za prikaz skeniranih dokumentov.

6.1.2 Napaka »Error in establishing connection«

Vzrok težave

Spletna aplikacija dostopa do storitve IMiS®/Capture Service preko domene, ki ni dovoljena v storitvi IMiS®/Capture Service.

Odprava težave

Uporabnik z administratorskimi pooblastili mora v storitvi IMiS®/Capture Service vnesti dovoljene domene iz katerih spletna aplikacija dostopa do storitve. Po vnosu mora ponovno zagnati storitev IMiS®/Capture Service. Po zagonu je potrebno osvežiti spletno stran brskalnika za prikaz skeniranih dokumentov.

6.1.3 Napaka »Socket connection error«

Vzrok težave

Spletna storitev IMiS®/Capture Service ni zagnana ali pa odjemalec nima pravic dostopa do naslova na katerem se nahaja IMiS®/Capture Service.

Odprava težave

Administrator mora ponovno zagnati storitev IMiS®/Capture Service. Po zagonu storitve osveži spletno stran brskalnika za prikaz skeniranih dokumentov.

6.1.4 Napaka »No scanner is connected«.

Vzrok težave

Ob zagonu IMiS®/Capture Service, storitev ni našla oziroma ji ni uspelo naložiti gonilnika za priključen skener.

Odprava težave

Administrator mora preveriti ali je skener vključen in povezan z računalnikom. Nadalje preveri, ali je aplikacija IMiS®/Scan zagnana. V kolikor je zagnana, jo mora zaustaviti in ponovno zagnati storitev IMiS®/Capture Service. Za več informacij glej [poglavje 4.2 Zagon in zaustavitev](#).

6.1.5 Napaka »Scanner: '{ime skenerja}' can not be loaded«.

Vzrok težave

Pred pričetkom skeniranjem storitvi IMiS®/Capture Service ni uspelo naložiti gonilnika za izbrani skener.

Odprava težave

Administrator mora preveriti ali je skener vključen in povezan z računalnikom. Nadalje preveri, ali je zagnana aplikacija IMiS®/Scan. V kolikor je zagnana, jo mora zaustaviti in ponovno poizkusiti s skeniranjem v aplikaciji IMiS®/wScan. Če tudi sedaj skeniranje ne deluje, mora ponovno zagnati storitev IMiS®/Capture Service. Za več informacij glej [poglavje 4.2 Zagon in zaustavitev](#).

6.1.6 Po uspešnem skeniranju ni možno nadaljevati skeniranja

Vzrok težave

Med skeniranjem je prišlo do prekinitve podatkovne povezave med skenerjem in računalnikom.

Odprava težave

Administrator mora ugasniti skener in ga odklopiti od delovne postaje. Ponovno mora prižgati skener in ga priključiti na delovno postajo. V aplikaciji IMiS®/wScan ponovno sproži postopek skeniranja. V kolikor skeniranje še vedno ne deluje naj odpre program »Task Manager«.

Med vsemi procesi poišče proces z imenom »fjctwsw.exe« ter ga prekine s klikom na gumb »End process«.

6.1.7 Med skeniranjem se prazne strani ne odstranjujejo

Vzrok težave

V nastavitvah profila niso nastavljene skenerske nastavitve za odstranjevanje praznih strani.

Odprava težave

Po nastavitvah profila v aplikaciji IMiS®/wScan, administrator zažene IMiS®/wScan administrativni modul in sledi postopkom opisanim v poglavju [4.3. Dodatne nastavitve](#).

6.1.8 Napaka »Your browser does not support Javascript ES6.

Update browser.«

Vzrok težave

Knjižnica »imis.scan.js« ne deluje v kolikor brskalnik ne podpira Javascript standarda ECMAScript6.

Odprava težave

Obstoječi brskalnik je potrebno nadgraditi na najnovejšo verzijo.

Opozorilo: Tudi najnovejša verzija Internet Explorer brskalnika ne podpira tega standarda.

6.1.9 Napaka »Your browser does not support WebSockets. Update browser.«

Vzrok težave

Knjižnica »imis.scan.js« ne deluje v kolikor brskalnik ne podpira WebSocket tehnologije.

Odprava težave

Obstoječi brskalnik je potrebno nadgraditi na najnovejšo verzijo.

Opozorilo: Tudi najnovejša verzija Internet Explorer brskalnika ne podpira tega standarda.