

Payer Barbara

A megújuló online folyóirat-kiadás¹

Tudásmegosztás, együttműködés, automatizálás

Az Open Journal Systems a tudományos folyóiratok online közzétételének és kezelésének nyílt forráskódú megoldása. Számos újítást vezet be, így a folyóirat-politika átláthatóbbá válik, és javul az indexelés. Az MTA Könyvtár és Információs Központ az OJS 3-as magyarított verziójával egy közös folyóirat-szerkesztő rendszert kínál, az open access jegyében térítésmentesen. A szolgáltatás előnye, hogy a különféle exportlehetőségekkel automatizmusokat tudunk kiépíteni a rendszerrel, például a DOI-regisztráció vagy az MTMT-be történő adatátadás terén. Mindezzel jelentős munkaórát takarítunk meg a szerkesztőségeknek, valamint a DOI-regisztrációval foglalkozó munkatársaknak és az MTMT adminisztrátoroknak. A szoftvert alkalmazó intézmények együttműködését, tapasztalatcseréjét egy közös tudásmegosztó felülettel segítjük és létrehoztuk a hazai OJS-felhasználók csoportját.

Tárgyszavak: folyóirat; kiadványszerkesztés, online üzemmód, szoftver; nyílt hozzáférés

Az Open Journal Systems mint folyóirat-megjelentetési és -szerkesztői rendszer

Az *Open Journal Systems* (OJS) egy olyan nyílt forráskódú webes rendszer, amelyet elsősorban tudományos, lektorált folyóiratok online megjelenítésére fejlesztettek ki, hogy automatizmusokon keresztül könnyebb kapcsolattartási lehetőséggel, tudásmegosztással és visszakereshető archiválási funkcióval segítsék a szerkesztőségi munkafolyamatokat.

Az OJS története

Az OJS fejlesztése 1998-ra nyúlik vissza, ekkor több más nyílt forráskódú szoftverrel² együtt kezdtek el fejleszteni a folyóiratok nyilvántartására, szerkesztésére és online megjelenítésére. A világ számos országában használnak ilyen³ vagy ehhez hasonló folyóirat-menedzselő szoftvereket, ezzel jelentősen csökkentve a kiadói költségeket, ugyanakkor növelve a publikációkhoz való hozzáférési lehetőséget.

A szoftver fejlesztése a *Public Knowledge Project* (PKP) keretében zajlik, ami egy több egyetem összefogásával létrejött kezdeményezés⁴. 1998-ban *John Willinsky* hozta létre a *University of British Columbia* nevű kanadai egyetemen, amihez később csatlakozott a *Stanford University* és a

Simon Fraser University Library. 2005 óta a PKP adminisztratív és operatív otthona a *Simon Fraser University Library*. A PKP a fejlesztések során három nagy partnerrel dolgozik együtt, amelyek jelentős anyagi és természetbeni támogatásokat nyújtanak a projekt számára: *Ontario Council of University Libraries*, *University of British Columbia Libraries*, *University of Pittsburgh Libraries*.

2001-ben jelent meg az OJS első verziója, amit 4 év múlva, 2005-ben követett az OJS 2.0 változat. A 3-as verzió kiadására 11 év elteltével, 2016. augusztusban került sor. Ez idő alatt a fejlesztők igyekeztek minden felhasználói visszajelzést megvizsgálni és a szükséges módosításokat beépíteni az új szoftverbe. A hosszas fejlesztés egy merőben új kinézetű és működésű OJS-rendszert eredményezett.

Az OJS 3 szoftverkövetelményei

A program a szabad szoftver keretei között készül, a *General Public Licence* (GNU) irányelvei mentén terjeszthető és módosítható.

Az *Open Journal Systems* webserveren üzemeltethető, rendszerigénye meglehetősen kicsi.

A cikk írásának idején a legfrissebb verzió az OJS 3.0.2, amely nyílt forráskódú rendszereken fut. A fejlesztők ajánlásai alapján PHP 5.3.7 vagy maga-

sabb verziójú futtató környezetet igényel, MySQL vagy PostgreSQL adatbázis-támogatással. Az adatbázisszerveren pedig célszerű minimum MySQL 4.1-es verziónak vagy PostgreSQL 9.1.5-ös verziónak lennie. Operációs rendszer tekintetében javasolják a UNIX alapú rendszereket, mint például a Linux, FreeBSD, Solaris, Mac OS X.

A rendszer telepítése során az adatok védelme érdekében fontos, hogy az OJS feltöltő könyvtára, ahova a szerkesztési folyamat során keletkező munkaanyagok és a megjelentetett cikkek kerülnek, a webről közvetlenül ne legyen elérhető, javasolt azt a szerver *www* mappáján kívül elhelyezni. A magasabb fokú biztonság érdekében a konfigurációs állományban engedélyezhető a regisztrációhoz a captcha szolgáltatás.

Egy vagy több folyóiratot kezelő platform – előnyök és hátrányok

Az OJS-rendszer alkalmas egy folyóirat (single journal) kiadásának támogatására, de képes akár több folyóirat (multiple journal) egymástól elkülönített kezelésére is. Ez utóbbi esetben egy összefoglaló főoldalon jelennek meg a rendszerben működő, nyilvánosan elérhető, akár bejelentkezés nélkül is olvasható folyóiratok, bejelentkezés után pedig megfelelő jogosultsággal elérhetőek a nem nyilvános folyóiratok is.

A szoftver más menüpontokat kínál fel, ha egyfolyóiratos vagy, ha többfolyóiratos rendszerként használjuk. Például egyfolyóiratos rendszer esetén nem jelenik meg az összefoglaló OJS főoldal.

Egy rendszerben több folyóirat

Előnyök:

- könnyebb karbantartani, menedzselni és frissíteni a rendszert;
- folyóiratonként egyedi megjelenés alakítható ki;
- mindegyik folyóiratnak szabványos webcíme van, amely egy választott rövidítéssel válik egyedivé.

Hátrányok:

- nehézkes az egyéni igények alapján, de minden folyóirat számára megfelelően átalakítani a rendszer működését;
- a nyelvi fordítások mindegyik folyóiraatra egységesen vonatkoznak;
- a tapasztalatok szerint az OJS által alkalmazott webcímtől eltérő címet nem lehet alkalmazni, mivel az megzavarja a metaadatok arathatóságát.

Minden, ami előny vagy hátrány egy többfolyóiratos rendszerben, annak az ellenkezője igaz az egyfolyóiratos OJS platformokra. Kevés számú folyóirat esetén, különösen, ha azok eltérő szakterületű, tudományos, lektorált, illetve nem tudományos folyóiratok, talán érdemes megfontolni a folyóiratonkénti külön OJS-telepítést. Több külön OJS-telepítés esetén az üzemeltetett folyóiratok számának növekedésével válik egyre nehezebbé a karbantartás és a frissítések követése. Véleményünk szerint, azoknál a folyóiratoknál, ahol biztosítani tudják az OJS-hez szükséges infrastruktúrát és hozzá a megfelelő szakembert és/vagy informatikust, ott érdemes a szerkesztőségnek saját OJS-t üzemeltetni, amelyhez segítséget nyújt a szoftver magyar nyelvű fordítása, számos magyar és angol nyelvű kézikönyv, videó tutorial⁵, Wiki, valamint az OJS *Hungarian Users Group* (HUG) közössége (lásd később).

Folyóirat-szerkesztőségi rendszer vagy folyóirat-megjelenítő felület

Az OJS webes alkalmazást kétféle funkcióra használhatjuk: bővített formában folyóirat-szerkesztő, -menedzselő és -megjelentető rendszerként, szűkített formában pedig csak a szerkesztő és menedzselő funkciók igénybevételével. Hosszú távon mindenképpen a teljes funkcionalitás igénybevétele a megtérülő, hiszen egy felületen tárolhatjuk az egyes számokkal, cikkekkel kapcsolatos valamennyi levelezést, bekért anyagot, bírálati véleményt. Mindez archiválható, visszakereshető. Azoknál a folyóiratoknál, ahol a szerkesztőbizottsági tagok a rendszeren kívül, e-mailben folytatnak párbeszédet az egyes cikkekről, ott a csatolt cikkváltozatokat nehéz egyben tartani és nyomon követni, az információk visszakeresése pedig szinte lehetetlen.

A tapasztalatok azt mutatják, hogy a hagyományos papíralapú szerkesztési munkát a szerkesztőségek nehezen váltják fel erre a modern, munkafolyamat-menedzselő rendszerre. Az átállásnál különösen nagy feladat hárul a folyóirat-menedzseri szerepkört ellátó személyre, mivel ő az, aki technikai segítséget nyújthat például a szerzőknek egy-egy cikkfeltöltésben, vagy a bírálóknak a bírálati szakvélemény elhelyezésében. Ennek ellenére léteznek olyan OJS-platformot használó szerkesztőségek, ahol a bírálói vagy lektori szakvélemények e-mailben érkeznek meg a főszerkesztőhöz vagy szerkesztőhöz, és azt a szerkesztő vagy maga a folyóirat-menedzser tölti fel a szerző, a bíráló vagy a lektor nevében.

Az optimális rendszerhasználat elsajátításához érdemes időt hagyni, eleinte a szoftvernek nem nyilvános formában csak a munkafolyamat-menedzselő oldalát kipróbálni. A gyakorlatban, amikor összeállt egy szám, esetleg döntés születik arról, hogy a szerkesztőség korábbi számokat is fel kíván-e vinni a rendszerbe, illetve, hogy engedik-e a felhasználóknak az oldalon az önregisztrációt, akkor lehet nyilvánossá tenni a folyóiratot. Itt érdemes megjegyezni, hogy csak az OJS-oldal főadminisztrátora tehet nyilvánossá egy folyóiratot.

OJS 2 kontra OJS 3 – koncepcionális és technikai váltás

A 2-es verziójú OJS-szoftver, köszönhetően a stabil, jól kidolgozott működésének, rendkívül népszerű lett, hosszú ideig a fejlesztők nem adtak ki újabb főverziót. Az évek alatt számos folyóirat tette az OJS 2-ben működő weboldalát egyedivé, alakította át a rendszert úgy, hogy az a legkényelmesebben kiszolgálhassa a saját szerkesztői tevékenységeit. A fejlesztői közösség segítségével nagyon sok bővítmény és modul készült hozzá, amely egyre robusztusabbá tette a központi OJS-szoftvert. Emellett egymással párhuzamosan a különböző országokban helyi fejlesztések is zajlottak, a lelkes programozók más nyílt forráskódú rendszerekkel (pl. Wordpress) való összekapcsolási módokat is elkészítettek, különféle megjelenítési sablonokat tettek közzé a 2-es verzióhoz. Mindez ahhoz vezetett, hogy bár egy éve megjelent a 3-as főverzió, mégis a hazai és a nemzetközi OJS-felhasználók többsége még mindig 2-es verziójú szoftvert használ, a 3-as verzióra történő váltás pedig annál nehezebb, minél jobban átalakították⁶ az eredeti kódsort.

Belepillantva a használati útmutatóba⁷ a 2-es verziójú OJS-nek a következő főbb tulajdonságai vannak:

- helyben installált és adminisztrálható rendszer,
- kialakíthatják a cikkek beküldési követelményeit,
- a szerkesztők maguk állíthatják be a szerkesztési folyamat egyes szakaszait,
- számos metaadat rendelhető a cikkekhez, például kulcsszavak,
- átfogó tartalomindexelés jellemzi,
- a Subscription modul segíti a számok késleltetett megjelentetését és a védett tartalmak kezelését,
- a Payments modul biztosítja a folyóirat-előfizetések kezelését,
- olvasói eszközöket biztosít a tartalmakhoz, amelyek egyedileg konfigurálhatók,

- az olvasók értesítéseket kérhetnek a folyóirat újdonságairól, valamint megjegyzéseket fűzhetnek egy-egy cikkhez,
- szöveggörnyezettől függő sűgöt kínál.

A legújabb főverzió megjelenésekor az elődje több mint 10 éves volt, ennyi idő alatt az internetes világ, a felhasználók igényei, a webes megjelenítők, mind hardveresen, mind szoftveresen jelentősen átalakultak. A fejlesztők ezt felismerve hozták létre az új, letisztult felületű 3-as verziót, amely mind megjelenésében, mind működésében alapvetően más, mint az elődje volt.

A fent felsorolt tulajdonságok közül a 3-as verzióban a cikk írása idején az alábbiak nincsenek még kialakítva:

- a Subscription (előfizetés, feliratkozás) modul jelenleg még nem elérhető, bár a 3.1-es változatban már remélhetőleg szerepelni fog;
- a Payments (fizetés) modul sem alakították még ki, így még nincs lehetőség a védett tartalmak, előfizetők számára történő megjelentetésére;
- a cikkekhez rendelt kulcsszavak egyelőre nem jelennek meg az olvasói felületen;
- több olvasói eszközt kihagytak a rendszerből, vannak, amelyek később sem fognak bekerülni.

Számos fejlesztés és javítás várhatóan a 3.1-es verzióban lesz benne, amelyet 2017 őszén adnak ki; erre a verzióra mindenképpen érdemes lesz frissíteni.

A fejlesztők az OJS 3.0 felhasználói kézikönyv elején a következő újdonságokra hívják fel a figyelmet:

- Egyszerűsített regisztráció:
A korábbi verzióhoz képest a regisztrációs felület is megújult, jóval átláthatóbb, kevesebb adatot tartalmaz. A fejlesztők az új verzióban folyamatosan dolgoznak az ORCID⁸ szolgáltatás és az OJS-regisztráció összekapcsolásán, hogy ezzel az azonosítóval egyértelműen meghatározható legyen a szerző.
- Rugalmas átjárhatóság a szerepkörök és a feladatok között:
A korábbi verzióhoz képest, ha egy felhasználó több szerepkörben is végez feladatokat egy folyóiratnál, akkor nem kell visszalépnie a főoldalra, hogy egy másik szerepkörhöz kapcsolódó feladatot elvégezhesen. A felhasználó számára elérhető funkciók kiválaszthatók a bal oldali menüsávról. Minél több jogosultsága van egy

felhasználónak, annál több beállítási funkciót ér el.

- Átalakítható szerepkörök:
Az OJS 3 alapértelmezett szerepkörei a következők:
 - Adminisztrátor: aki létrehozza a folyóiratokat a rendszerben, és elvégzi a teljes OJS-re vonatkozó különféle beállításokat. Ilyen jogosultságot csak egy felhasználó kaphat több folyóiratot kezelő platform esetében is, és az adatait már a telepítés során meg kell adni.
 - Folyóirat-menedzser: csak folyóiraton belül létező szerepkör, de ott a legmagasabb jogosultságokkal rendelkezik. Több felhasználó is megkaphatja ezt a jogosultságot. Rálátása van az egész OJS-rendszerre, kereshet a többi folyóirat felhasználói között és azokból felhasználókat vehet át.
 - Szerkesztő: valamennyi szerkesztési munkafolyamatra rálát, anélkül, hogy részese vagy felhasználója lenne az adott cikk szerkesztési fázisának.
 - Rovatszerkesztő: amennyiben rovatokat alakít ki a folyóirat, úgy szükség lehet a rovatszerkesztőre. Ha a rovatok beállításánál előre megadjuk a rovatszerkesztőt, akkor ő minden cikkhez automatikusan hozzá van rendelve, amit az adott rovathoz küldenek be.
 - Bíráló: külső bírálókat is fel lehet kérni, és a szerkesztőségen belül is lehet bíráló szerepkört kiosztani a felhasználóknak. A cikkhez felvett bírálók alapértelmezés szerint csak a bírálati szakaszhoz férnek hozzá.
 - Lektor: felkérés alapján a számára kijelölt cikk lektorálását végzi el.
 - Tördelőszerkesztő: a kézirat tördelését, publikálható formátum kialakítását végzi. A formátum többféle lehet, PDF, HTML, XML stb.
 - Korrektor: a tördelt változat ellenőrzésére felkért személy. Az előállítási szakaszban egy párbeszéd keretében töltheti fel a javítási javaslatait, amely alapján a tördelőszerkesztő javításokat végez, majd a javított változatot újra fel kell töltenie.
 - Szerző: a szerzői jogokkal rendelkező felhasználó csupán kéziratot küldhet be a folyóirathoz. Fontos, hogy ha a cikk nem a szerző által vagy nem a nevében kerül feltöltésre, akkor nem tud hozzáférni a szerkesztési folyamathoz.
 - Olvasó: azok a felhasználók, akik olvashatják a közzétett cikkeket, értesítést kaphatnak, ha megjelenik egy folyóiratszám.

Fontos megjegyezni, hogy a bíráló, lektor, tördelőszerkesztő, korrektor csak akkor férhet hozzá a számára kijelölt szakaszban a cikkhez, ha az adott szerepkörrel hozzáadták a cikkhez és egy felkérő (értesítő) e-mailt kap a feladatról.

Az OJS szerepkörei jogosultság szerint hierarchikusan lefelé átjárhatóak, vagyis az adminisztrátor, szerkesztő és rovatszerkesztő szerepkörökkel be lehet jelentkezni egy, az adott szerepkörnél alacsonyabb jogosultságokkal rendelkező felhasználó profiljába. Így lehetőség van a rendszeren belül egy-egy felhasználónak segítséget nyújtani vagy nevében eljárni, ha elakad (például egy szerzőnek a cikkfeltöltésben).

Az OJS 3-ban a szerepkörökhöz tartozó jogosultságok átállíthatók, ezzel szabályozható, hogy melyik szerepkör felhasználója melyik szerkesztési szakaszhoz férhet hozzá. Emellett természetesen új, egyedi szerepkörök is létrehozhatók.

- Egy cikkhez több fájl feltöltésének lehetősége:
Az új felületen a szerző több kéziratot tartalmazó fájlt is feltölthet a cikkhez, szemben az OJS 2-es verziójával, ahol tételenként csak egy dokumentumot nyújthatott be. A kéziratokhoz továbbra is kapcsolhatók egyéb mellékletek: képek, táblázatok, diagramok stb. Korábban élesen elkülönültek ezek a különféle dokumentumok, a jelenlegi működés pedig ezt a megkülönböztetést próbálja árnyalni azáltal, hogy minden objektumot a fő benyújtás részeként kezel.
- Szerkesztői megbeszélések:
Az OJS 3-as verziója egy belső kommunikációs felületet kínál, annak érdekében, hogy a cikk útja a benyújtástól a megjelentetésig – vagy az esetleges elutasításig – nyomon követhető legyen. A későbbi visszakereshetőség szempontjából fontos, hogy ezek a szerkesztői párbeszédetek egy helyre kerüljenek, és ne csak az egyes szerkesztőségi tagok e-mail postafiókjában legyenek meg – ha egyáltalán ott fellelhetők.

Egy belső párbeszéd elindításához mindig legalább két személy kell, de igény szerint bármennyi felhasználó bevonható a beszélgetésbe. A résztvevők e-mailben értesítést kapnak, hogy meghívták őket a szerkesztési folyamat valamelyik párbeszédébe.

– Rugalmas munkafolyamat:

Az OJS 3-ban mindössze 4 szakaszból áll egy cikk életútja a benyújtástól a megjelentetésig:

- 1. szakasz: a beküldött cikk területe, ahol feltölthető a kézirat, és láthatók a szerző által megadott metaadatok. Fontos megjegyezni, hogy többszerzős cikk feltöltése során, a feltöltőnek meg kell határoznia, hogy melyik szerző lesz a fő kapcsolattartó a szerkesztési munkafolyamat során; ez a személy nem feltétlenül egyezik meg a feltöltővel. A cikk valamennyi szerzőjét fel kell venni egy-egy külön úrlapon, amely űrlap látszólag hasonlít az OJS adminisztrációs oldalán lévő felhasználói űrlaphoz, de attól jóval kevesebb adatot tartalmaz, és az itt felvitt neveket és e-mail címeket a rendszer nem hasonlítja össze a korábban regisztrált felhasználókkal. Ennek megfelelően nem okoz gondot, ha egy szerző több folyóiratnál, vagy több cikkben is szerző, mellette esetleg még regisztrált felhasználója a rendszernek. Ebben a szakaszban a cikkel csak akkor lehet további tevékenységeket végezni, például áthelyezni a bírálati szakaszba, ha a cikkhez hozzá van rendelve legalább egy szerkesztő vagy rovatszerkesztő.
- 2. szakasz: bírálati rész, ahol több körben, egy vagy több bíráló bevonásával zajlik a cikk bírálati folyamata. A bíráló számára határidők állíthatók be, először a felkérő e-mailre a válasz határideje, hogy a felkért bíráló elvállalja-e vagy sem a bírálatot. Pozitív válasz esetén pedig a bírálat leadásának a határidejét kell betartania bírálónak. A rendszerből e-mail útján lehet figyelmeztetést küldeni a határidők betartására. A szükséges számú bírálat beérkezését követően határozhat a szerkesztőség arról, hogy változtatás nélkül megjelenteti-e a cikket – ami ebben az esetben átkerülhet a nyelvi lektorálás szakaszba –, esetleg változtatásra kéri a szerzőt vagy szerzőket, majd a változtatás után új bírálati folyamat is indulhat, illetve el is utasíthatja a kéziratot. Fontos figyelni arra, hogy vak bírálat vagy kettős vak bírálat esetén a kéziratból és a fájl tulajdonságai közül is ki kell törölni a szerzőre vonatkozó információkat, míg a bírált fájlból ugyanígy a bírálóra vonatkozó utalásokat. A szerkesztőség bírálati gyakorlatától függően (kettős-vak, vak vagy nyílt bírálat) itt is kezdeményezhető párbeszéd.
- 3. szakasz: nyelvi lektorálási rész, amely egy hasonló munkafolyamat, mint a bírálat, annyi különbséggel, hogy alapértelmezés szerint az ide bekerült kéziratok már elfogadott státusz-

ban vannak, meg fognak jelenni a lapban. A nyelvi lektor kijelölése előtt a szerkesztőnek fel kell töltenie a bírálaton átesett kéziratot ebbe a szakaszba. A nyelvi lektor pedig a javított példányt tölti fel ugyanide. Természetesen ennél a szakasznál is kezdeményezhető párbeszéd, akár a szerző bevonásával is.

- 4. szakasz: a megjelentetés fázisa, ahol a tördelőszerkesztő bevonásával zajlik a folyóirat számára megfelelő publikálási formátum (pl. PDF vagy HTML) előállítás. A tördelőszerkesztő külső, saját szoftverrel végzi el a tördelést, majd feltölti a kész anyagot a „Megjelenítő létrehozása” lehetőséggel. Ebben a szakaszban történhet a tördelést követő korrektúrázás is, a korrektúrázó személyének hozzáadásával, valamint a korábbi szakaszoknak megfelelően itt is kezdeményezhető párbeszéd a szerzővel, vagy más szerkesztőségi taggal. Ha a cikk tartalma és formátuma megfelelő, akkor besorolható valamelyik korábban létrehozott folyóiratszámába, ezáltal ütemezhető a megjelentetése. A szerkesztési folyamat rugalmasságát mutatja, hogy a kézirat könnyen áthelyezhető egyik fázisból a másikba, akár bizonyos fázisok ki is hagyhatók.

– Elkülönített stíluslapok:

Az új rendszer az olvasói felületet és az adminisztrációs felületet elkülönítetten kezeli, ezáltal egy egységes adminisztrációs felületen kezelhető a folyóirat, míg a publikus oldal alkalmazza a folyóirat számára kidolgozott stílust és megjelenítési formát. Az átdolgozott stíluslap későbbi módosítása jóval egyszerűbb, kevesebb munkával jár.

– Reszponzív megjelenés, Bootstrap technológia alkalmazása:

Az OJS 2-es verzió megjelenése a több mint 10 évvel ezelőtti technológiai eszközökhöz, számítógépekhez, böngészőkhöz igazodott. Az elmúlt években jelentős technológiai változás ment végbe, a felhasználók új, megváltozott méretű és felbontású eszközökön interneteznek, igény jelentkezett a rezponzív OJS-szoftverre. A fejlesztők az OJS 3-ban már a Bootstrap technológiát alkalmazták, amely segíti a folyóiratonként egyedi külső kialakítását. Az alapértelmezett sablonon csak azokat az elemeket kell módosítani, amelyeknél változtatást szeretnénk elérni. Ezenkívül más lehetőségek is kínálkoznak az OJS 3-ban tárolt folyóirat egyedi megjelené-

sének kialakítására, ezek leírására a cikk későbbi részében kerül sor.

Az MTA KIK OJS 3 multijournal platformja

Az MTA Könyvtár és Információs Központban az OJS 3-as verzióját választottuk, mivel a 2-es verzió frissítése hamarosan befejeződik, az új pedig igyekszik eleget tenni napjaink technológiai követelményeinek. Azonban azzal is szembe kellett néznünk, hogy mint minden új szoftvernek, így ennek a 3-as verzióknak is számos olyan pontja van még, amely nem működik tökéletesen. A fejlesztők igyekeznek ezeket kijavítani, de néha a szerkesztőségeknek kompromisszumot kell kötniük, legalább a javított verzió kiadásáig.

Az MTA KIK OJS-rendszerét a cikk írásának idején 11 különböző témájú folyóirat használja, egy részük igénybe veszi a szerkesztőségi és a megjelenítési funkciót is, többségük csak szerkesztőségi rendszerként alkalmazza.

A szolgáltatás elindításához szükséges feladatok és lépések

A munkát 2016 tavaszán kezdtük, ekkor még nem jelent meg a 3-as verzió, és nem is tudtuk, hogy pontosan mikor fogják majd kiadni. Így először a 2-es verziót kezdtük el tesztelni. Már a munka elején fontosnak tartottuk, hogy a szoftver magyar felületen működjön, készüljön hozzá magyar nyelvű kézikönyv – legalább az elérhető angol nyelvű kézikönyvnek a magyar fordítása. Így készült el először a 2.3.3-as angol nyelvű kézikönyv magyar változata, valamint az akkori legfrissebb verzió, a 2.4.8-as szoftver magyarítása. Ezek elkészítése és a rendszer tesztelése eltartott a 3-as verzió megjelenéséig, 2016. nyár végéig.

Az új verzióról szóló beharangozó hírek csábítóak voltak, így fontosnak tartottuk, hogy a rendszerünk éles bevezetése előtt telepítsük az új verziót és hasonlítsuk össze az előzővel. Az új felülettel kapcsolatos első benyomások nagyon meggyőzőek voltak, így felmerült, hogy mégis az új verzióval induljunk el, ehhez azonban még több tesztre és a fejlesztői fórum rendszeres olvasására volt szükség, hogy lássuk, vannak-e problémás, még nem működő részek, amelyek nélkül az új verziót egyelőre nem tudjuk használni. Pár hetes tesztelést követően, egyeztetve az OJS használatára nálunk igényt tartó folyóiratok szerkesztőségeivel, nem

találtunk olyan negatívumot a rendszerben, amely miatt el kellett volna vetnünk az új verziót. Ennek megfelelően 2016 novemberében az OJS 3-as verzióját ajánlottuk fel használatra az érdeklődő szerkesztőségeknek. Mellette megtartottuk teszt jelleggel a 2-es verziót is, olyan esetekre, amikor más OJS-felhasználókkal kell együttműködnünk (például rekordok repozitóriumi bearatása vagy DOI-regisztráció kapcsán). A 3-as verzió honosítását a 2-es változat alapján már könnyen el tudtuk készíteni, a korábbi fordítást az új programrészek kiegészítésével bővítettük.

A tesztelési szakasztól kezdve számos bemutatót, előadást tartottunk. Ezek a bemutatók nem csak azoknak a szerkesztőségeknek szólnak, amelyek az MTA KIK OJS-rendszerét szeretnék használni, azoknak is szívesen segítünk, akik saját szerveren kívánják telepíteni és használni az OJS-rendszert. Számos szerkesztőségnek küldtük el a szoftver magyar fordítását.

A magyar nyelvű felület elkészítése

A 2-es verzióhoz hasonlóan a 3-as változatban is XML kiterjesztésű állományok a nyelvi fájlok, amelyek a rendszer két fő területén találhatók:

Vocale{az adott nyelv kódja}

Vib{pkpVocale}{az adott nyelv kódja}

A nyelvi kódot pedig a **registryVocales.xml** állományban rögzíthetjük, itt található a rendszerből elérhető valamennyi nyelv. Ide kell egy új sort rögzíteni és a megfelelő paraméterekkel ellátni. Fontos, hogy az itt megadott kulcs (key) paraméter megegyezzen a fenti elérési útvonalaknál jelzett nyelvi kóddal. Például a magyar nyelv esetében: **hu_HU**.

Az XML fájlok létrehozása történhet az angol nyelvi xml-ek lemásolásával, majd átszerkesztésével. A fájl elején a **<locale>** tag paraméterének átírásával kell kezdeni.

Erről:

```
<locale name="en_US" full_name="U.S. English">
```

erre:

```
<locale name="hu_HU" full_name="Hungarian">
```

majd le kell fordítani a **<message key="">** tag-ek értékeit.

Előfordulhat, hogy a létrehozott új nyelv felületén bizonyos feliratok helyett ilyen vagy ehhez hasonló karaktorsor jelenik meg –

##admin.settings.siteLogo## – ami azt jelenti, hogy az adott nyelvi környezetben nem található a feliratnak megfelelő message key. Ez a jelenség a leggyakrabban szoftverfrissítéskor jelentkezik, amikor új funkciók lépnek be a rendszerbe, amit be kell építeni a helyileg hozzáadott nyelvi változatba is. Ennek a legegyszerűbb megoldása, ha a friss angol nyelvi változathoz (esetleg valamely szoftver karakter összehasonlítási funkciójával) átmásoljuk a hiányzó szövegeket és magyarítjuk.

Az MTA KIK a 3.1-es verzió megjelenése után tervezi, hogy teljessé teszi a magyar nyelvi fordítást és a fejlesztőkkel felveteti a szoftverbe ezt a nyelvi csomagot is mint alapértelmezett magyar nyelvi változatot. Ez természetesen a későbbiekben folyamatos humánerőforrást és fordítási követést igényel, mert minden szoftverfrissítésnél – közreműködve a fejlesztőkkel –, a magyar nyelvi változatot is frissíteni kell.

A fő nyelvi fájlok mellett ide kell venni a pluginok nyelvi állományát, valamint a helyzetérzékeny súgót is. Az előbbieket a **plugins** mappán belül az egyes bővítmények könyvtárainak **locale** almappáiban találhatók. A súgó nyelvi fájljai nem XML állományokban vannak, hanem **md** kiterjesztésű állományokban a **docsmanual[adott nyelv kódja]** szerint. A magyar nyelvű súgót a **hu** nevű könyvtárba kell elhelyezni, ahol a tartalomjegyzék a **SUMMARY.md** állományban található, az egyes fejezetek pedig a fejezetcím alapján azonosíthatók.

A rendszer felületének és automatizmusainak elfogadtatása a szerkesztőségekkel

A tapasztalatok azt mutatják, hogy az OJS használatának elsajátítása számos nehézséggel járhat. A visszajelzések szerint azok számára, akik megszokták a 2-es verziót, az új változat átláthatatlan, bármilyen letisztult is a felülete. A teljesen új felhasználóknak pedig az online folyóirat szerkesztési koncepcióját kell megérteni, adaptálni, esetleg a korábbi szerkesztési folyamataikat az online munkafolyamatokhoz alakítani. Az elmúlt 8 hónapos aktív használat alatt számos új funkciót tapasztaltunk, az alábbiakban ezek közül emelünk ki néhányat:

Feladatok: bejelentkezés után a rendszerben lévő szerepköreinknek megfelelően egy helyen láthatjuk a feladatainkat. Ez egy egyszerű, a feladat kiosztásának megfelelő időrendi felsorolás, ami egy nagyobb folyóiratnál lévő szerkesztő számára, vagy egy több folyóiratban is szerkesztőként szereplő személynek nehezen átlátható. A felsorolás

csak a cikk címét tartalmazza, amihez kapcsolódik a feladat, illetve a folyóirat nevének rövidítését. A felsorolásban a cikk címére kattintva léphetünk el magához a feladathoz, például egy bírálati felkérés esetén a cikk bírálati szakaszához. Szükség esetén a feladatok olvasottá tehetőek (félkövér betűszedés helyett normál betűvel jelennek meg a listában) valamint törölhetőek.

Beküldések: a korábbi verzióktól eltérően más bontásban jelennek meg a beküldött kéziratok. Nem egy oldalon, hanem három külön lapon találhatóak, de nem a feldolgozási stádiumuk szerint:

- az aktuális felhasználóhoz rendelt beküldések oldala;
- az aktuális, még meg nem jelent cikkek oldala;
- archívum, a megjelent cikkek oldala.

Az adott lapokon belül egy konkrét feldolgozási szakasz (pl. bírálathoz) cikkeit szűrővel tudjuk leválogatni.

Új beküldés: a rendszer lehetőséget kínál arra, hogy más nevében végezzünk el bizonyos feladatokat. Ahhoz, hogy a szerzőt automatikusan hozzárendeljük egy cikkhez, mindenképpen vagy magának a szerzőnek, vagy a szerző nevében kell a cikket feltölteni. Ez lehetővé teszi, hogy a szerző láthassa a szerkesztési folyamatot, valamint a megjelentetett cikknél a valódi szerző neve jelenjen meg. Fontos megjegyezni, hogy a rendszer akkor is küld visszaigazolást a szerzőnek, hogyha a szerző nevében töltötték fel a kéziratot. Ezt érdemes előre egyeztetni a szerzővel, hogy tudja, neki mi a teendője ebben az esetben.

Megbeszélések: a szerkesztési folyamat különböző szakaszaiban lehetőség van megbeszélést (párbeszédet) kezdeményezni a szerzővel, vagy a szerkesztőség egy vagy több tagjával. A bíráló, lektor, tördelőszerkesztő stb. számára küldött felkérő leveleket a rendszer szintén párbeszédnek tekinti és ott is jeleníti meg. Fontos megjegyezni tehát, hogy a bírálathoz, lektoráláshoz stb. felkért személyeket nem elegendő felvenni a cikk résztvevői közé, hanem a Notify/Értesítés segítségével értesíteni is kell a felkérésről. Ameddig ez nem történik meg, nem férnek hozzá az adott kéziratához.

Archívum: a rendszerben megjelenő esetleges duplumbeküldéseket a felület csupán egyszer, a beküldés során a kézirat feltöltésének felületén jelezheti, amennyiben a fájl neve alapján egyezést talál egy korábbi beküldéssel. Ez a jelzés azonban könnyedén figyelmen kívül hagyható, akár két ugyanolyan cikket egymástól függetlenül el lehet juttatni a megjelentetési szakaszba, amikor (a Schedule For Publication/Publikálás ütemezése segítségével) hozzárendeljük a cikkeket a következő folyóiratszámhoz. Az így hozzárendelt cikkek

automatikusan megjelennek az adott szám tartalomjegyzékében. Természetesen lehetőség van utólag törölni a tartalomjegyzékből a cikket, de ettől az még az Archives/Archívum szakasznál Published/Publikált, vagyis megjelentetett stádiumban marad, ami a későbbiekben már nehezen kezelhető és értelmezhető. Az ilyen duplumokat érdemes a szerkesztési folyamatból is törölni. A törlés ebben az esetben nem keverendő össze a cikk elutasításával, hiszen az elutasított cikkek az Archives/Archívum területre kerülnek, Decline/Visszautasított státusszal, míg a törölt cikkek visszavonhatatlanul eltűnnek a rendszerből.

Bírálati kör: a rendszerben korlátlan számú bírálati kör hozható létre, ahova felvehető a kézirat éppen aktuális változata. Az egyes körök létrehozásánál nem vizsgálja a rendszer, hogy az előző kört megfelelően lezárták-e. Mindig a legutolsó bírálati körnél tudjuk a cikket kezelni, döntést hozni annak további sorsáról. A jelenleg aktuális 3.0.2-es verzióban a véletlenül létrehozott bírálati köröket nem lehet a felületen törölni, csak az adatbázisból, figyelembe véve a *submission.current_round* értéket, valamint a *review_rounds* és a *review_round_files* táblákban lévő, a cikkhez tartozó bejegyzéseket.

Felhasználói szerepkörök: folyóirat szinten a legnagyobb jogosultsággal rendelkező felhasználó a folyóirat-menedzser (Journal Manager), aki beállításokat végezhet az adott folyóiraton belül, segítséget nyújthat az ott lévő felhasználóknak, bejelentkezhet a nevükben, hogy hozzáférhessen a felhasználó felületéhez. Fontos megjegyezni, hogy azoknál a felhasználóknál, akiknek ugyanabban az OJS-rendszerben egy másik folyóiratnál is van valamilyen szerkesztési joggal rendelkező szerepkörük, ott nincs lehetősége a folyóirat-menedzsernek belépni a felhasználó felületére. Mivel az OJS egy felhasználót csak valamelyik folyóirathoz rendelten tud kezelni, így előfordulhat, hogy egy felhasználó már a rendszerbe történő regisztrációja során nem megfelelő folyóirathoz kerül, vagy nem kerül be egyetlen folyóirathoz sem. Ebben az esetben a folyóirat-menedzser ugyan meg tudja keresni a felhasználót, ha a keresés során jelöli, hogy a rendszer valamennyi felhasználója között keres, azonban a jelenlegi verzióban csak manuálisan, az adatbázisban lehet a felhasználót a folyóiratok között mozgatni.

Szintén működési sajátosságként figyelhető meg, hogy egy konkrét cikknél nem lehet valaki egyszerre bíráló és nyelvi lektor, vagy bíráló és rovatszerkesztő. Ezeket a tevékenységeket külön-külön felhasználói fiókkal érdemes elvégezni.

A többnyelvű felület egységes kezelése: ha a folyóiratnál több nyelvet engedélyezünk, akkor valamennyi nyelv esetében egységesen be kell állítani az arculati elemeket (logót, színvilágot, borítóképet stb.). A többnyelvű űrlapoknál minden nyelven meg kell adni az adatokat, mivel ennek hiányában a webaratási funkciók, XML formátumú exportok hiányos adathalmazt fognak eredményezni.

Visszamenőleges feldolgozás: azoknál a szerkesztőségeknél, ahol korábban megjelent folyóiratszámokat szeretnének feldolgozni és megjelentetni az OJS felületén, hasznos eszköz lehet a QuickSubmit plugin, amely elérhető már a 3-as verzióban is. A bővítmény segítségével egy felületen keresztül már publikálásra kész cikkeket tölthetünk fel a rendszerbe a szükséges metaadatok megadásával. A feltöltés során lehetőség van a fájlt a szerkesztési területen elhelyezni, vagy azonnal publikálni.

Az egységes egyediesség paradoxon

A rendszer által használt e-mail sablonszövegek folyóiratokként módosíthatók, akár saját sablonszövegek is készíthetők. A szoftver nyelvi fordításai viszont az egész rendszerre hatással vannak. A több folyóiratot kezelő rendszerben az egyik legnehezebb feladat, hogy megtartsuk a rendszer egységességét, de mégis eleget tegyünk az egyes szerkesztési politikából fakadó kívánságoknak. Ismerünk olyan szerkesztőségeket, ahol több tucat folyóiratot külön-külön OJS-rendszerben működtetnek. Ennek hátránya, hogy minden frissítést és központi változtatást valamennyi rendszerben át kell vezetni. Nagyobb humán erőforrást igényel, ha több különálló OJS-rendszerben folyamatosan szeretnénk követni a szoftveres frissítéseket és egyéb változtatásokat. Egy több folyóiratot működtető OJS-ben végzett módosításnál javasolt minden folyóirattal először egyeztetni, hogy a változtatás minden szerkesztőség számára elfogadható legyen, például egy link vagy egy gomb felirata, esetleg egy űrlap feldolgozása után megjelenő üzenet a felhasználónak.

Amikor a felhasználók nemcsak felületi, hanem működésbeli módosítást szeretnének, az három okból is átgondolást igényel:

- a szoftver frissítések elvesznek a módosítások;
- a különféle folyóiratok számára vajon elfogadható lesz-e az új működés;
- az OJS-ben tárolt adatok arathatósága nem sérül-e.

Tapasztalataink szerint azokban az esetekben, ahol a működés nem megfelelő, nem felhasználóbarát vagy nehézkes, érdemes az OJS hivatalos fórumát is átolvasni, valamint megnézni a bevezetésre váró frissítéseket a PKP hivatalos oldalán⁹. Célravezetőbb a nem túl súlyos hibáknál a fejlesztői frissítést megvárni, mint egyénileg módosítani a rendszert. Ha mégis módosítás szükséges, akkor két nagyon fontos szabályra kell figyelni:

- mindig valamilyen tesztrendszeren kell elvégezni a módosítást;
- az összes rendszerbeli folyóirattal érdemes előre egyeztetni a várható új működést, esetleg tesztelési lehetőséget biztosítani a folyóiratok számára is.

Az egyedi megjelenés kialakításának lehetőségei

A szerkesztőségek számára mindig fontos, hogy a felület, ahol megjelennek, mennyire formálható a saját arculatukra. Az OJS korábbi verziójában látszólag könnyebb az oldalt egyedivé tenni, módosítani a stílusfájlokat, talán emiatt is láthatunk annyira sokféle és egyedi OJS 2-es felületet. Ezzel szemben az új verzióban a Bootstrap technológia alkalmazása a jellemző, ami két fő részből áll: egy központi sablonból, valamint a folyóiratonkénti leszármaztatott sablonokból (Child-theme). Ezeket a leszármaztatott sablonokat az OJS 3 bővítményként kezeli, tehát minden olyan elemet létre kell hoznunk, aminek a segítségével a sablonunk, mint bővítmény telepíthető és aktiválható lesz.

Ez utóbbi elkészítésén kívül természetesen akadnak további módszerek, amelyekkel átalakítható az alapértelmezett megjelenés; a következőkben ezekről lesz szó.

Template (.tpl) fájl átírása

A **templates** mappában található template fájlok (.tpl) módosítása sokszor a legegyszerűbb megoldásnak tűnik, hogyha szeretnénk a megjelenített felületeket átalakítani. A template fájlok többféle elemet is tartalmazhatnak: HTML, PHP, valamint különféle formázó stílusokat. Az alábbiakban egy részlet látható az **article_details.tpl** fájlból, amelynek segítségével megjeleníthetjük a cikkhivatkozásokat:

```
{if $article->getCitations()}
<div class="item references">
  <h3 class="label">
    {translate key="submission.citations"}
  </h3>
  <div class="value">
    {$article->getCitations()}|nl2br}
  </div>
</div>
{/if}
```

A hivatkozások megjelenítésének (akár folyóiratonkénti) szabályozására a rendszer a jelenlegi verzióban nem kínál lehetőséget. Az OJS fórumán az ezzel kapcsolatos igények és hibajelzések is megtalálhatók. A fejlesztők válasza szerint egy későbbi verzióban javítani fogják ezt a szolgáltatást. Ameddig ez az ígért fejlesztés nem elérhető, lehetnek olyan folyóiratok, amelyek nem szeretnének a cikk alatt hosszú hivatkozáslistát megjeleníteni. Ebben az esetben kézenfekvő megoldásnak látszik a *tpl* kiterjesztésű fájlból eltávolítani a fenti sorokat. Az egyszerű eltávolítás azonban kihat a teljes rendszerre, ami egy több folyóiratot működtető OJS-ben nem engedhető meg.

Ha mindenképpen szükséges így belenyúlni a forráskódba, akkor a törlés helyett érdemesebb a megjelenítést feltételhez kötni. Szerencsére az OJS-ben a legtöbb elemre (pl. cikk, folyóirat száma, szerző, aktuális felhasználó stb.) hivatkozhatunk. Amennyiben a fenti kódot közrefogjuk az alábbi feltétellel, akkor megadhatjuk, hogy melyik folyóiratnál jelenjen meg a hivatkozási lista a cikkek végén és melyeknél nem:

```
{if $article->getContextId() != '11'}
{/if}
```

Általánosságban azonban elmondható, hogy a template fájlok ilyen jellegű módosítása nem szerencsés, hiszen egy frissítés során nehéz ezeket a módosításokat mindig átvezetni az új rendszerre. További nehézség lehet, hogy a szerver fájljaihoz való hozzáférés általában korlátozott, folyóiratmenedzserek számára nem mindig engedélyezett.

Egyedi CSS fájl alkalmazása

Az OJS 3-as felületén lehetőségünk van egyedi CSS fájl feltöltésére, amiben felülbírálni lehet a

rendszer központi sablonját. Ebben az esetben kihasználható a Bootstrap technológia előnye, vagyis nincs szükség minden stíluselem felvételére, csak azokat kell megjelenítenünk a stílusfájlban, amelyeket módosítani szeretnénk.

Például egy cikkabsztraktra az **obj_article_details.abstract** tag segítségével hivatkozhatunk, a láblécben megjelenő fejlesztői logóra pedig a **pkp_brand_footer** taggel.

Az OJS által alkalmazott stíluscímkék felderítésére hasznos eszköz lehet a Mozilla Firefox Firebug kiegészítője, vagy a Google Chrome böngészőben az oldal forráskódját megjelenítő és elemző eszköz. A szükséges stíluselemeket kell egy új CSS fájlba másolni, majd a stílushoz tartozó jellemzőket tesszés szerint módosítani. A kész CSS fájl az adminisztrációs felületen keresztül tölthető fel a Settings/Website/Appearance (Beállítások/Weboldal/Megjelenés) részben. Egyedi CSS fájl készítése általában kisebb felületi átalakítások során lehet hasznos.

Child theme készítése

A fejlesztők minden folyóirathoz egyedi sablont, ún. child theme-t javasolnak létrehozni, amelyhez hasznos útmutatót és egy mintát is kínálnak¹⁰. Az OJS 3-as verziójában a megjelenítéshez létrehozott sablonokat bővítményként kezeli a rendszer, ezért az alapértelmezett téma a *plugins/themes/default/* mappában található, amely mellé a többi származtatott sablont közvetlenül a szerverre kell feltölteni a *plugins/themes* mappa folyóiratonkénti almappáiba. Ezek olyan sablonok, amelyek kibővítik az alapértelmezett sablon tulajdonságait stílusokkal, szkriptekkel. Fontos, hogy a mappa nevében szerepeljen az alábbi karaktorsor: **_child**. A bővítményként felismert sablon a későbbiekben bármelyik folyóiratnál alkalmazható.

A child theme mappa az alábbiakból áll:

```
img/  
locale/  
styles/  
index.php  
version.xml  
ChildThemePlugin.inc.php
```

Az *index.php* hívja meg a *ChildThemePlugin.inc.php* fájlt, amelyben többek között azt definiálhatjuk, hogy be legyen-e töltve az alapértelmezett téma, és mellé a saját stílusfájlunk, vagy kizárólag a saját stílusfájlunk alapján építse-e

fel az oldalt a rendszer. Minden esetben kötelező a *version.xml* fájl, ami a témáról alapvető információkat tartalmaz. A könyvtárak elnevezései alapján látható, hogy melyikbe milyen fájlok valók, így a képeknek szolgáló mappa az *img*, a *locale* a nyelvi XML-eket tartalmazó mappa, nyelvenként külön almappákkal. A stíluskönyvtárba kerül a megjelenést meghatározó fájl, amely ebben az esetben nem CSS típusú, hanem LESS kiterjesztésű állomány. A LESS fájl valójában egy kibővített CSS, amit egy értelmező program, például jelen esetben a PHP lefordít, és átalakít CSS fájlá.

Részlet egy LESS fájlból:

```
@var: #004590;  
div {  
  height: 50px;  
  width: 50px;  
  background-color: @var;  
  &:hover {  
    background-color: fadeout(@var, 50%)  
  }  
}
```

Részlet egy CSS fájlból:

```
div {  
  height: 50px;  
  width: 50px;  
  background-color: #004590;  
}  
  
div {  
  background-color: rgba(0, 69, 144, 0.5);  
}
```

A szerkesztőségeknek figyelembe kell venniük, hogy az egyedi megjelenés kialakításához bővebb informatikai ismeret szükséges, emellett bizonyos esetekben közvetlen szerverhozzáféréssel lehet a kész sablonállományt feltölteni.

Együttműködés és tudásmegosztás

OJS 3 platform nyílt hozzáférésű tudományos folyóiratok részére

Az MTA Könyvtár és Információs Központ elkötelezett a nyílt hozzáférésű technológiák iránt, felismerte az Open Journal Systems hazai terjedésének ütemét, a megváltozott folyóirat-kiadási tendenciákat. Célja, hogy összefogja a hazai OJS-felhasználókat, térítésmentesen OJS-platformot nyújtson azoknak a tudományos, nyílt hozzáférésű folyóiratoknak, amelyek önerőből nem tudják megoldani egy ilyen rendszer üzemeltetését.

Információs oldal a folyóirat-kiadók és OJS-t használók számára

Az MTA KIK multijournal OJS-platformjának bevezetése óta számos olyan dokumentum keletkezett, amely hasznos lehet a hazai OJS-felhasználók számára, ezért létrehoztunk egy olyan információs oldalt¹¹, amely egy felületen jeleníti meg az eddig keletkezett dokumentumokat és segédanyagokat. A legfontosabb ilyen segédanyag az OJS 2-es és az OJS 3-as verzió magyar nyelvű kézikönyve.

Oktatások és nyomtatható segédanyagok

Az MTA KIK igyekszik az OJS-rendszer használatával kapcsolatos valamennyi tapasztalatát megosztani, átadni, amivel nemcsak a könyvtár OJS-platformját használó folyóirat-szerkesztőseket kívánja segíteni, hanem más OJS-felhasználókat is. Ennek jegyében több helyszínen tartottunk oktatásokat és számos, a rendszer bevezetésével kapcsolatos első iránymutató megbeszélésen vettünk részt.

Az OJS-rendszerben leggyakrabban alkalmazott munkafolyamatokról képes, rövid útmutatókat is készítettünk, amelyek szintén letölthetők az OJS-hez készített információs oldalunk *Felhasználói kézikönyvek* részéből.

Fórum a rendszerrel kapcsolatos tapasztalat-cseréhez

Hazánkban egyre népszerűbb az online folyóirat-kiadás, a megjelentetéshez egyre több szerkesztőség az OJS-szoftvert veszi igénybe. Erre a tendenciára reagálva az MTA Könyvtár és Információs Központ létrehozta a magyarországi OJS-felhasználók közösségét (OJS Hungarian Users Group), amelybe meghívtuk azokat a partnereket, akik már OJS-felhasználók, vagy akik tervezik a rendszer bevezetését, függetlenül attól, hogy az MTA KIK OJS-oldalát kívánják-e használni, vagy saját rendszert üzemeltetnek.

Az MTA KIK-ben az OJS bevezetésével járó tapasztalatok alapján, valamint a hazai OJS felhasználói csoport szervezése miatt szükségesnek éreztük egy fórum kialakítását. Célunk, hogy az OJS-ben szerzett tapasztalatokat minél szélesebb körben meg tudjuk osztani.

A fórum féléves működése alatt kevés aktív beszélgetés látható, mivel a felhasználók jellemzően közvetlenül a könyvtár OJS-adminisztrátorát kere-

sik meg a kérdéseikkel. Ennek ellenére a könyvtár igyekszik azokat a témafelvetéseket, hibajelzéseket, beállításokat kiírni a fórumra, amelyek nagyobb közönség számára is érdekesek és hasznosak lehetnek, mint például az online megjelenő közlemények egyértelmű azonosítására és hosszú távú elérésük biztosítására szolgáló DOI-azonosító plugin engedélyezése és beállítása¹² az OJS két fő verziójában.

Automatizálás az egységes adatátadás és webaratás érdekében

Az MTA KIK célja egy olyan szabványos adatimportáló eszköz kifejlesztése, amelynek segítségével pár kattintással elhelyezhető az OJS-ben megjelentetett cikk a könyvtár REAL repozitóriumában, valamint a Magyar Tudományos Művek Tárában (MTMT). Az adatimportáló szoftver működése az MTMT2-es szoftverhez és annak táblastruktúrájához igazodik.

A könyvtár megrendelésére az *MTA SZTAKI Elosztott Rendszerek Osztály* munkatársai kifejlesztettek egy OJS közös keresőt¹³, amely kereshetővé teszi azokat az OJS-ben tárolt folyóiratcikkeket, amelyek metaadatai az OAI PMH protokoll segítségével lekérdezhetők. Az Open Archives Initiative Metaadatgyűjtő Protokoll (OAI PMH¹⁴) egy olyan szabványos keretrendszer, amely a metaadatok begyűjtésén alapul, jelen esetben az OJS-rendszerektől, mint adatgazdától gyűjti össze egy keresőfelületre a folyóiratcikkek metaadatait.

Az OJS 2-es rendszerben külön beállítás szükséges, hogy az OAI PMH lekérdezés sikeres legyen, a 3-as verziójú OJS-ben ez a szolgáltatás alapértelmezett, beállítást nem igényel.

Úton az ideális publikációs platform felé

A tanulmányban vázolt működési jellemzők rávilágítanak arra, hogy az OJS 3 egy rendkívül rugalmasan alakítható, egyéni igényeknek megfelelően testreszabható rendszer. A munkafolyamatok számos ponton kapcsolódnak, így bár a szerkesztési lépések általában visszavonhatók, de előfordulhat, hogy több lépésben, több felületen kell különféle beállításokat végezni ahhoz, hogy egy adott művelet sikeres legyen, és ne maradjanak cikktörédek vagy felesleges bejegyzések a rendszerben. Látható, hogy az alig egyéves szoftverben vannak még hibák, hiányosságok, de a nemzetközi OJS-közösség aktivitása alapján vár-

hatóan a 3-as verzió is egy stabil, jól működő rendszerré fogja kinőni magát.

Hivatkozások

- ¹ A cikk a 2017. április 19-21. között megrendezett Networkshop 2017 konferencián elhangzott előadás bővített változata.
- ² Open Conference Systems (OCS): <https://pkp.sfu.ca/ocs/> Open Monograph Press (OMP): <https://pkp.sfu.ca/omp/>
- ³ PKP Index: <http://index.pkp.sfu.ca/index.php/browse/index/all>
- ⁴ <https://pkp.sfu.ca/about/>
- ⁵ <http://pkpschool.sfu.ca/>
- ⁶ Átalakított nemzetközi OJS oldal: <http://jmhs.cmsny.org/index.php/jmhs>
Átalakított hazai OJS oldal: <http://tet.rkk.hu/index.php/TeT>
- ⁷ http://openaccess.mtak.hu/dokumentumok/OJS/ojs_2_3_3_felh_utm.pdf
- ⁸ www.orcid.org
- ⁹ https://github.com/pkp/pkp-lib/milestones?direction=asc&sort=due_date&state=open
- ¹⁰ <https://github.com/NateWr/default-child>
- ¹¹ <http://openaccess.mtak.hu/index.php/fejlesztok/ojs>
- ¹² <http://forum.mtak.hu/viewtopic.php?f=6&t=7&sid=9dc33ef41099fc68a38363037d71c74c#p16>

<http://forum.mtak.hu/viewtopic.php?f=6&t=8&sid=9dc33ef41099fc68a38363037d71c74c#p17>

¹³ <http://oai.kereso.sztaki.hu/kereso/index.php?type=1>

¹⁴ <http://www.openarchives.org/OAI/openarchivesprotocol.html>

Irodalom

Public Knowledge Project: <https://pkp.sfu.ca/> [Utolsó elérés: 2017. szeptember 7.]

Public Knowledge Project Community Forum: <https://forum.pkp.sfu.ca/> [Utolsó elérés: 2017. szeptember 7.]

Public Knowledge Project OJS Documentation: https://pkp.sfu.ca/wiki/index.php?title=OJS_Documentation [Utolsó elérés: 2017. szeptember 7.]

Learning OJS 3 : A visual Guide to Open Journal Systems: <http://openaccess.mtak.hu/dokumentumok/OJS/ojs3-en.pdf> [Utolsó elérés: 2017. szeptember 7.]

Open Archives Initiative: <http://www.openarchives.org/> [Utolsó elérés: 2017. szeptember 7.]

Beérkezett: 2017. X. 11-én.



Payer Barbara

az MTA Könyvtár és Információs Központ Szakinformatikai Osztályának munkatársa.

E-mail:

payer.barbara@konyvtar.mta.hu