

## Egy olvasmány-, illetve forráslista-kezelő rendszer létrehozása

A tanárok által a hallgatóik számára összeállított annotált könyv- és cikklisták elterjedt elemei a brit felsőoktatásnak. Manapság ezek már többnyire elektronikus formában készülnek és a hagyományos információhordozók mellett online tartalmakra mutató linkeket is tartalmaznak. Az ilyen ajánlott forrásokat és olvasnivalókat tartalmazó listák létrehozására, karbantartására és szolgáltatására fejlesztették ki a *Resource/Reading List Management System (RLMS)* elnevezésű szoftvereket.

Egy RLMS beüzemelése gondos tervezést igényel, ha azt szeretnénk, hogy sikeres legyen. Az első lépés az oktatási intézmény döntéshozóinak meggyőzése arról, hogy szükség van egy ilyen rendszerre, ami amellet, hogy segíti a diákokat a tanuláshoz szükséges irodalom és egyéb források megtalálásában, a könyvtárnak is hasznos útmutató lehet ezek beszerzéséhez. Egy RLMS arra is lehetőséget nyújt, hogy a tanulók visszajelzéseket küldjenek az olvasmánylisták hasznosságáról, továbbá hogy a könyvtár figyelemmel kíséresse ezek használatát. A felső vezetés mellett természetesen meg kell nyerni a listákat összeállító egyetemi oktatók és az azokat használó diákok támogatását is a tervezés és a megvalósítás során egyaránt, és szükség lehet a számítóközpont szakembereinek bevonására is, hogy az RLMS kompatibilis legyen az intézmény egyéb rendszereivel, például a VLE-vel, vagyis a virtuális tanulókörnyezettel.

A tervezési fázisban nem árt néhány kérdést tisztázni, amelyek befolyásolhatják, esetleg megnehezíthetik a megvalósítást. Ilyenek például: az olvasmánylisták tulajdonjogai (az intézmény vagy az őket összeállító oktatók ezek tulajdonosai? vagy szabályozatlan a jogi helyzetük?); a listák száma (minden modulhoz vannak ilyenek? léteznek olyanok is, amelyek nem kötődnek egy konkrét modulhoz?) és átlagos hosszúsága (a műszaki szakterületeken jellemzően kevés ajánlott irodalmat adnak meg, míg például a társadalomtudományi szako-

kon a több száz tételes jegyzékek sem ritkák); az adatrögzítés felelősei (maguk a tanárok, a tanszéki adminisztrátorok, a könyvtárosok, vagy (fizetett) külső munkaezők fogják bevinni ezeket a listákat? vagy lehet, hogy már jelentős részük rendelkezésre áll valamilyen elektronikus formában?); a listák karbantartása (ki fogja javítani és bővíteni az adatokat? ugyanazok, akik eredetileg gépre vitték őket, vagy esetleg a könyvtárosok csinálják az elsődleges adatbevitelt, de a frissítést már az oktatók vállalják?); a hozzáférés kérdése (csak az intézmény tagjai használhatják az RLMS-ben tárolt listákat? vagy legyenek nyilvánosak és bárki megtekinthesse őket?).

Egy RLMS-től ilyen alapfunkciók várhatók el:

- A diákok amellet, hogy megnézhetik a listákat, az ajánlott dokumentumok elérhetőségét is egyszerűen ellenőrizhetik.
- A listák létrehozásának és karbantartásának lehetősége az arra jogosult személyekre korlátozható.
- Az oktatók igényeik szerint átszerkeszthetik és annotálhatják a listákat.
- A listák importálhatók és exportálhatók (pl. bibliográfia-kezelő szoftverekből, illetve ilyenekbe).
- A könyvtárosok értesítést kapnak a változásokról, az új vagy törölt tételekről (segítve ezzel az állományfejlesztést).

Elvárás lehet még az oktatási intézményben működő más – például autentikációs, kurzuskezelő, könyvtári – rendszerekkel való integrálhatóság vagy adatcsere lehetősége is.

Létezik néhány fizetős RLMS szoftver, illetve szolgáltatás, továbbá vannak nyílt forráskódú ingyenes rendszerek is, és egyre több a *freemium* üzleti modellt követő megoldás (ezeknél vagy csak az alapfunkciók ingyenesek, a speciálisak már nem, vagy például a fejlesztők az olvasmánylistákból indított vásárlások után jutalékot kapnak az online

könyvesboltoktól). [1] A kész rendszerek közötti választást nagyban segíti, ha kikérdezzük azok használóit a tapasztalataikról, az esetleges problémákról. További lehetőség lehet a saját fejlesztés, amivel a helyi igényeknek leginkább megfelelő és a többi lokális adatbázissal jól integrálódó rendszer alakítható ki, de érdemes végiggondolni előbb egy ilyen saját készítésű szoftver költségeit is, és természetesen azzal is számolni kell, hogy a fejlesztésre szükséges idő miatt csak később lehet majd használatba venni a rendszert – különösen, ha menet közben újabb és újabb igények, ötletek merülnek fel (*feature creep*).

Egy RLMS rendszer bevezetése a következő főbb fázisokból áll: az érintettek támogatásának megszerzése, a rendszer telepítése és konfigurálása, a felhasználók képzése, az adatbevitel, és végül a felhasználók támogatása. Ha van rá idő, akkor hasznos lehet előbb egy *pilot* verzió beüzemelése (pl. csak egy vagy néhány egyetemi kar bevonásával), mert ez a tesztelés mellett arra is alkalmas, hogy ezeknek a feladatoknak egy részében gyakorlatot szerezzenek a résztvevők.

Ahhoz, hogy minél több olvasmány- és forráslista bekerüljön a rendszerbe, meg kell győzni a tanszékeket a leendő szolgáltatás fontosságáról – akár úgy is, hogy az egyetem vezetése eldöntse, hogy minden kurzushoz vagy modulhoz szükséges ilyen listákat készíteni és be kell őket tenni az RLMS adatbázisába. A meggyőzés mellett arról is érdemes tájékozódni a könyvtárosoknak, hogy hogyan kezelik, és hol jelentetik meg az oktatók ezeket a listákat? milyen módon hívják fel rájuk a hallgatóik figyelmét és mivel motiválják őket a használatukra? van-e segítségük a jegyzékek összeállításában és karbantartásában? Ha az a döntés született, hogy az RLMS-be nem a tanszéki dolgozóknak kell az adatokat bevinniük, akkor valószínűleg könnyebben megnyerhetők az ügytámogatóknak, mert ilyenkor az oktatóknak csupán a listákat kell átadniuk az adatrögzítéshez, más feladatuk nincs kezdetben.

Az RLMS szoftver installálása természetesen az intézményi informatikusok vagy a rendszert szállító cég szakembereinek feladata. A legproblémásabb rész a többi helyi rendszerrel való összekapcsolás lehet. A következő lépés az intézményi adatok feltöltése: például a szervezeti felépítés és a kurzusok/modulok adatai – ezeket jó esetben importálni lehet egy már létező másik nyilvántartásból.

A rendszer használatának megtanítása és a felhasználók folyamatos támogatása szintén fontos eleme a sikernek, különösen akkor, ha a listák bevitelét a tanszékeknek kell megoldaniuk. Hasznos lehet oktatóvideót vagy online tananyagot készíteni, amit bármikor elő lehet venni, ha új embert kell betanítani. A használat közben felmerülő problémák megbeszélésére egy telefonszámot vagy e-mail címet kell biztosítani, és esetleg egy blogot vagy fórumot is lehet indítani (utóbbiaknál egy „Gyakran Ismételt kérdések” szekció is nagyon hasznos tud lenni).

Sok szempontból a legkritikusabb munkafázis az olvasmánylisták adatainak bevitele. Ha nem sikerül a kitűzött indulási határidőre elérni a „kritikus tömeget”, akkor fennáll a veszélye, hogy nem veszik majd komolyan az új szolgáltatást és az egész projekt halálra van ítéelve. Ezért folyamatosan ellenőrizni kell a feltöltési ütemet, és ha az elmarad a tervezettől, akkor vagy ösztönözni az adatrögzítőket, vagy további embereket bevonni, vagy visszavenni az eredeti célkitűzésből és például először csak az elsőéveseknek készült listákat vinni be a rendszerbe. A minőségbiztosításra is oda kell figyelni: korrektek-e az irodalomjegyzék egyes tételei, illetve rendben van-e a lista felépítése? A típushibákat érdemes a felhasználóképzés tananyagába belevenni, így remélhetőleg a jövőben csökkenni fog a gyakoriságuk.

Egy 90%-os, vagy annál nagyobb feltöltöttségi arány a szolgáltatás nyilvánossá tételekor már igazán jó eredménynek számít. De ahhoz, hogy idővel a még hiányzó listák is bekerüljenek az adatbázisba, és hogy a már felvitt adatok karbantartása, frissítése is folyamatos legyen, az elindulás után is folytatni kell a szervezési és népszerűsítési munkát. Utóbbit ilyenkor már a tulajdonképeni célcsoportra, vagyis a hallgatókra is ki kell terjeszteni, hogy elkezdjék használni ezt az új szolgáltatást.

## Hivatkozás

[1] [http://helibtech.com/Reading\\_Resource+lists](http://helibtech.com/Reading_Resource+lists)

**/BREWERTON, Gary: Implementing a Resource or Reading List Management System. = Ariadne, 71. sz. 2013./**

(Drótos László)