SLUB
Wir führen Wissen.

Request for an offer

„Software development for the FOLIO Library Service Platform (LSP) - Computing and visualization of E-Resource usage statistics“

We hereby ask for an offer for the development of software for the FOLIO Library Service Platform (LSP) for the calculation and visualization of usage statistics for e-resources (see specification below). Send the offer until 30 October 2020 to haushalt.einkauf@slub-dresden.de with reference to the ID M351-2020.

Use references to proof your experience in agile software development and with public institutions. Use software projects on GitHub to proof your experience in developing open source software.

Please briefly describe the extent to which the personnel to be appointed has experience with the same or a similar software stack. Indicate whether the personnel to be appointed has already contributed to software development in the FOLIO LSP project and briefly describe the contributions.

Please provide an estimate of the workload in working days as well as the daily rate (net) and any other costs incurred and indicate whether the deadline can be met.

Client

Saxon State and University Library Dresden (SLUB)
Postal Address: 01054 Dresden
Visitors' Address: Zellescher Weg 18, 01069 Dresden

Subject

Software development for the FOLIO Library Service Platform -
Computing and visualization of E-Resource usage statistics

Background

SLUB is one of the largest research libraries in Germany. As a classical state library it archives and collects comprehensively publications on Saxony and publications published in Saxony. As university library of the Technische Universität Dresden it guarantees the information supply to a strong research university with an unusually wide range of subjects. Furthermore it fulfills important coordination and service functions to the libraries in the Free State of Saxony.

SLUB operates the Dresdner Digitisation Center (DDZ), a leading center for mass digitization, and is a member of the competence network German Digital Library. It participates substantially in the development of production and presentation software of digital documents.

In the field of resource discovery, SLUB cooperates closely with the Leipzig University Library. Both operate a joint article index with more than 150 sources and 160 million records, which are also made available to other libraries within the finc user community[1].

SLUB is a public institution of the Free State of Saxony.

In partnership with the Leipzig University Library, SLUB is conducting a project to supply the Saxon university libraries with an Electronic Resource Management System based on the open source FOLIO Library Service Platform[2] (LSP). This project is funded by the European Regional Development Fund (ERDF)[3] and ends in December 2020. FOLIO is an international collaboration of libraries, developers and vendors building an open source library services platform. It supports traditional resource management functionality and can be extended to other institutional areas. Since May 2018 the Leipzig University Library has been a member of the Open Library Environment (OLE) community and is actively involved in the development of the FOLIO LSP. One of the project results is the development of the FOLIO app *eUsage* for automatic harvesting and storing COUNTER[4] usage statistics of electronic resources. The majority of functional requirements for Electronic Resource Management are almost in place due to the performance of the community. However, there are some business-critical functions that should be realized by commissioning a software company. These functionalities are described below.

Goal

The outcome of this project should enable FOLIO libraries and their staff to evaluate licensed electronic resources by means of calculated and visualized statistics. These statistics will base on the COUNTER statistics releases 4 and 5 within the FOLIO Library Service Platform. This involves further classes of information such as agreements, journal and/or package information as well as payment information (cross-app use cases).

A number of visualized statistics will be available in the existing FOLIO app "Agreements", which can be downloaded for further use. The application administrator can configure in a client-specific setting which of the available statistics are available to the employees of the library.

The code / app has to to be open source under the terms of the Apache License, Version 2.0.

The project goal coincides with the goals of the international FOLIO community. It will benefit from this development.

---

[1] https://finc.info/

[2] https://www.folio.org/

[3] https://ec.europa.eu/regional_policy/en/funding/erdf/

[4] https://www.projectcounter.org/

Specification

Overview and system context

The development in this project is part of the FOLIO LSP, which consists of the OKAPI API gateway[5] and apps, maintained by various development teams and stakeholders from all over the world. It is to be assumed that the development will take place in coordination and cooperation with parts of this community. The objectives are specified by SLUB; the atomic requirements analyses are coordinated and provided by the Leipzig University Library, which also acts as product owner within the FOLIO community.

More information on FOLIO software development, references and guidelines can be found at https://dev.folio.org/. It is imperative to follow the FOLIO software development rules and guidelines. All software components are to be distributed under the terms of the Apache License, Version 2.0. The FOLIO project uses the agile management method SCRUM, which implies bi-weekly sprints, sprint review and planning.

The software stack is composed as follows: Docker, React, REST, PostgreSQL, JSON, Java, Javascript, Git. Please consult https://dev.folio.org/ for more detailed information

The following FOLIO apps are important for the goals set out here:

Agreements: FOLIO app, used for the administration of agreement information as well as e-resources, i.e. journal package and journal information. Package and journal information can be obtained via the Global Open Knowledgebase (GOKb) using the app Local KB Admin, or via EBSCO knowledge base, using the app eHoldings. The choice can be made at tenant level.

The app Agreements is compliant to both knowledgebases and stores the information on licensed content as agreement lines, linking to the knowledgebase app.

- folio-org/mod-agreements: Module for handling the server side work needed for the Agreements app[6]
- folio-org/ui-agreements: Stripes front end module for Agreements app[7]

eUsage: FOLIO app, used for the record of Sushi access credentials, COUNTER release version and report types as well as the storage of COUNTER usage stats according the COUNTER data model of releases 4 and 5.

- folio-org/mod-erm-usage: Store usage statistics and access data to these statistics[8]
- folio-org/ui-erm-usage: Stripes UI for managing usage statistics[9]

---

[5] https://github.com/folio-org/okapi/

[6] https://github.com/folio-org/mod-agreements

[7] https://github.com/folio-org/ui-agreements

[8] https://github.com/folio-org/mod-erm-usage

[9] https://github.com/folio-org/ui-erm-usage

Orders: FOLIO app, where orders and order lines are managed.

- folio-org/mod-orders: Orders business logic module[10]
- folio-org/mod-orders-storage: Storage (CRUD) module for Orders[11]
- folio-org/ui-orders: Stripes UI for managing orders[12]

Invoices: FOLIO app, where invoices are managed.

- folio-org/mod-invoice: Invoice business logic module[13]
- folio-org/mod-invoice-storage: Persistent storage (CRUD) of invoice data
- folio-org/ui-invoice: Stripes UI for invoices
-

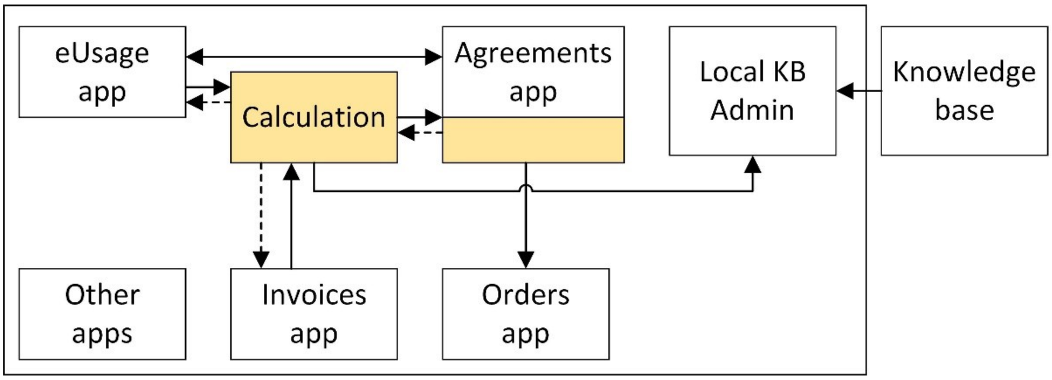
The complete FOLIO source code is available on https://github.com/folio-org/.

System Context Diagram

The system context diagram shows involved apps and classes of information.

The white boxes represent already existing FOLIO apps. The yellow boxes represent apps/components/parts that are to be developed in this project. There are two different Knowledgebases interoperating with the FOLIO System which both manage package and e-journal information: the Global Open Knowledgebase (GOKb) and the EBSCO Knowledgebase. Both are to be examined and - if technically feasible - supported with regard of the project goals.

The statistical calculations, that have to be implemented in this project, use data from the apps Agreements, eUsage, Invoices and Orders, as listed above. Results have to be visualized in the agreements app and have to be prepared for download for further use (see section Example scenario below).



Example scenario “Cumulative journal access per time”

The scenario “Cumulative journal access per time” is exemplary for a number of scenarios that are to be developed. It is intended to be used for illustration. The scenarios mainly differ in the type of

---

[10] https://github.com/folio-org/mod-orders
[11] https://github.com/folio-org/mod-orders-storage
[12] https://github.com/folio-org/ui-orders
[13] https://github.com/folio-org/mod-invoice

statistics, which information sources (app modules) are included, how they are calculated and visualized.

Goal:

A library employee wants to know how often the journals that are licensed under a particular agreement have been used in their library. He uses the FOLIO agreements app for this. He opens the agreement and selects the "Cumulative journal access per time" statistics from the available options in the "Statistics preview" section. He selects the reporting period. The default period is the date from the start to the end of the agreement (or until now if the contract is still running). Then he selects "Show statistics" and sees a commented diagram that shows the number of uses (y) per time in months (x). He can then select "Download" and receives a table with the numbers on which the diagram is based and the diagram itself.

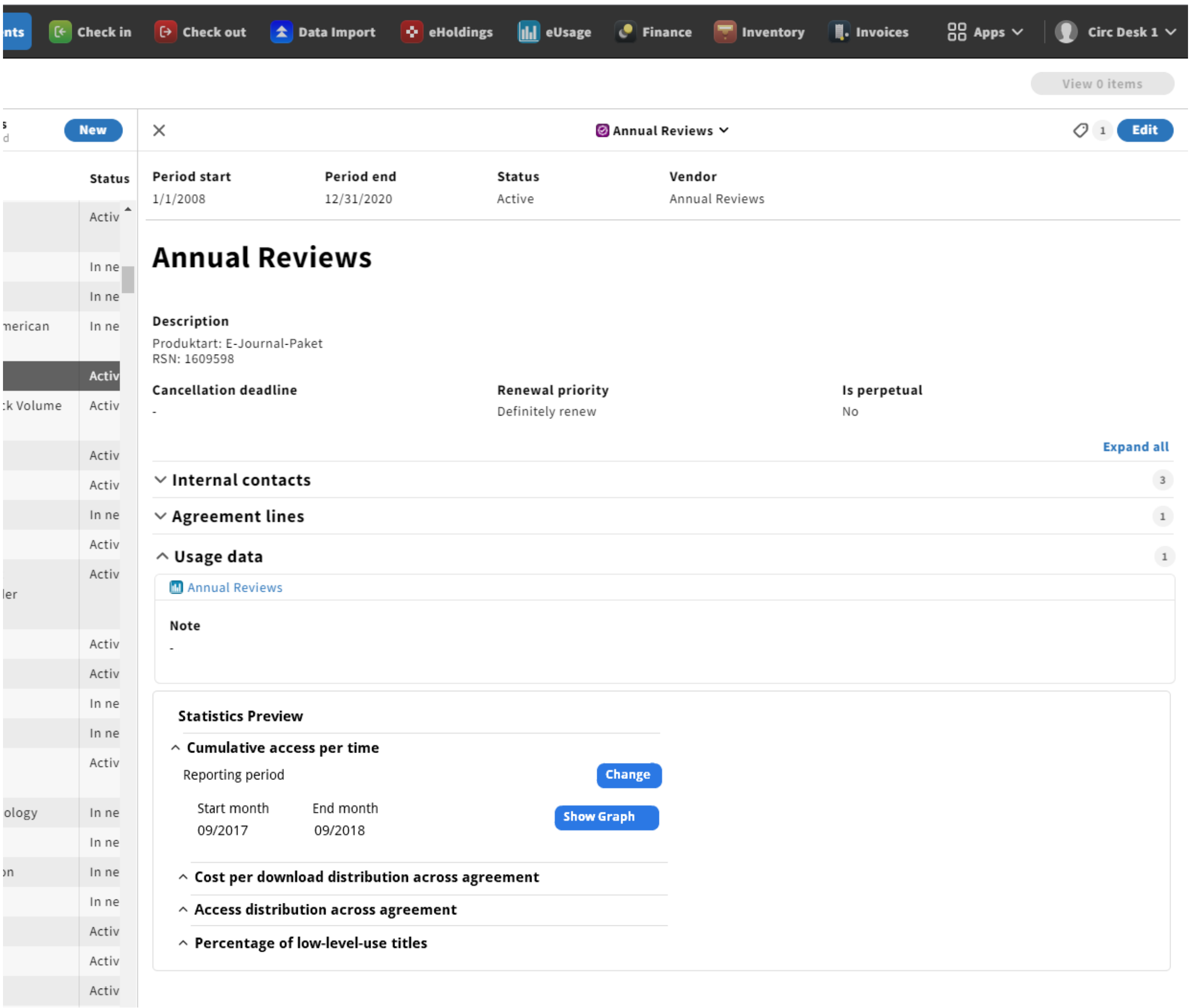
Scenario steps (screenshots do not represent the current state of the FOLIO software and include some mock-ups):

(1-USER)

- User selects an agreement in agreements app.
- User opens the usage data section on the agreement screen and selects the statistics “Cumulative journal access per time”.

(2-USER)

- User enters reporting period start and end month.
- User clicks on “Show graph”.



Period start 1/1/2008
Period end 12/31/2020
Status Active
Vendor Annual Reviews

Annual Reviews

Description
Produktart: E-Journal-Paket
RSN: 1609598

Cancellation deadline -
Renewal priority Definitely renew
Is perpetual No

Internal contacts
Agreement lines
Usage data
Annual Reviews

Note
-

Statistics Preview
Cumulative access per time
Reporting period
Start month 09/2017
End month 09/2018
Cost per download distribution across agreement
Access distribution across agreement
Percentage of low-level-use titles

(3-SYSTEM)

- Parameters:
  - statistics ID (not yet present)
  - agreement ID
  - usage data provider ID
  - reporting period start month
  - reporting period end month
- The Statistics ID defines the code to be executed and thus all further steps. The further steps and parameters therefore only apply to the "Cumulative journal access per time" statistics. Parameters:
  - required report version: 4
  - required report type: JR1

(4-SYSTEM)

- The program collects additional data from other modules through APIs.
  - folio-org/mod-erm-usage

        - in: usage data provider ID, report version: 4, report type: JR1, reporting period start month, reporting period end month
        - out: JSON file containing JR1 report for the reporting period
    - folio-org/mod-agreements
        - in: agreement ID
        - out: JSON file containing entitlements list for the agreement (KBART-ish list of ISSNs/DOIs and other data for all journals covered by the agreement)

(5-SYSTEM)

- The program reports lines from mod-erm-usage according to entitlements list (only those journals remain in the report, that are listed in the entitlements and therefore are currently subscribed).

(6-SYSTEM)

- The program sums up the full text access counts of each month for all journals.

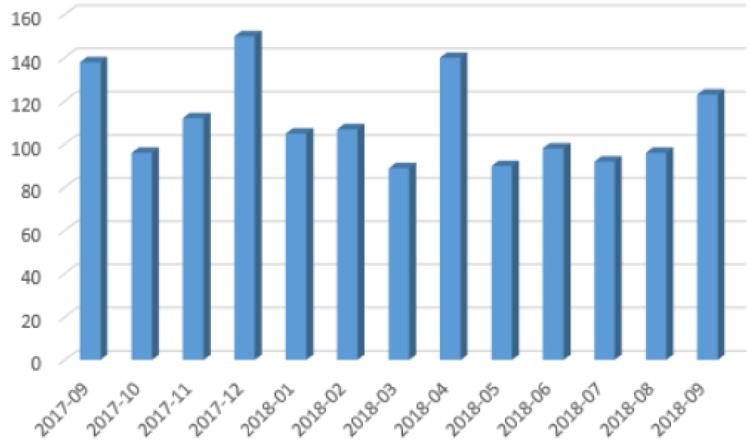
(7-SYSTEM)

- The program makes data-table exportable via API as JSON:

Month	Full text access
2017-09	138
2017-09	96
2017-09	112
2017-09	150
2017-09	105
…	

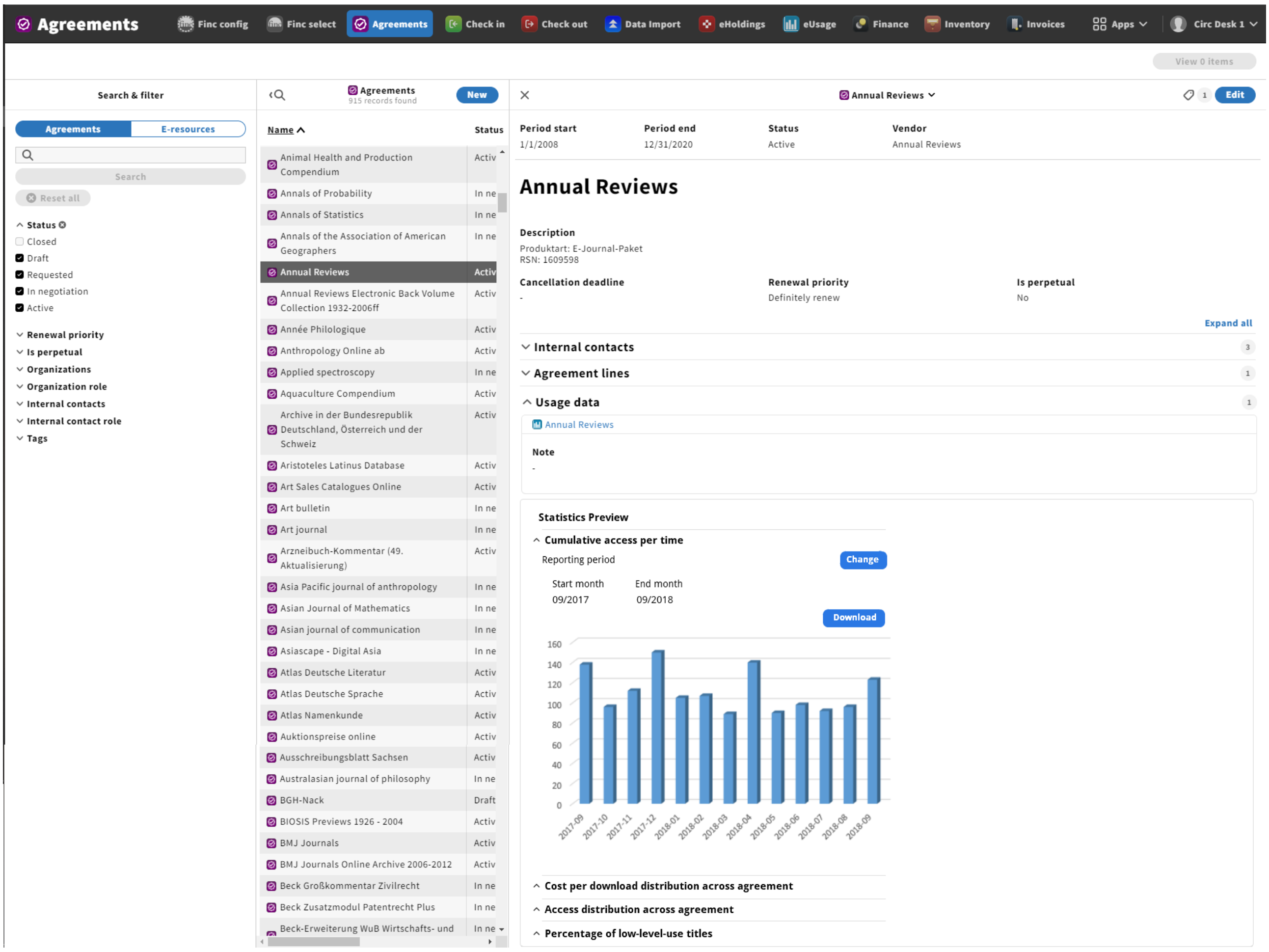
(8-SYSTEM)

- The program generates a graph from the data table and makes it exportable via API as SVG.



(9-SYSTEM)

- The program displays the graph in the agreements user interface.



Status
- ☐ Closed
- ☑ Draft
- ☑ Requested
- ☑ In negotiation
- ☑ Active

Renewal priority
Is perpetual
Organizations
Organization role
Internal contacts
Internal contact role
Tags

Agreements
915 records found

Name	Status
Animal Health and Production Compendium	Activ
Annals of Probability	In ne
Annals of Statistics	In ne
Annals of the Association of American Geographers	In ne
Annual Reviews	Activ
Annual Reviews Electronic Back Volume Collection 1932-2006ff	Activ
Année Philologique	Activ
Anthropology Online ab	Activ
Applied spectroscopy	In ne
Aquaculture Compendium	Activ
Archive in der Bundesrepublik Deutschland, Österreich und der Schweiz	Activ
Aristoteles Latinus Database	Activ
Art Sales Catalogues Online	Activ
Art bulletin	In ne
Art journal	In ne
Arzneibuch-Kommentar (49. Aktualisierung)	Activ
Asia Pacific journal of anthropology	In ne
Asian Journal of Mathematics	In ne
Asian journal of communication	In ne
Asiascape - Digital Asia	In ne
Atlas Deutsche Literatur	Activ
Atlas Deutsche Sprache	Activ
Atlas Namenkunde	Activ
Auktionspreise online	Activ
Ausschreibungsblatt Sachsen	Activ
Australasian journal of philosophy	In ne
BGH-Nack	Draft
BIOSIS Previews 1926 - 2004	Activ
BMJ Journals	Activ
BMJ Journals Online Archive 2006-2012	Activ
Beck Großkommentar Zivilrecht	In ne
Beck Zusatzmodul Patentrecht Plus	In ne
Beck-Erweiterung WuB Wirtschafts- und	In ne

Period start	Period end	Status	Vendor
1/1/2008	12/31/2020	Active	Annual Reviews

Annual Reviews

Description
Produktart: E-Journal-Paket
RSN: 1609598

Cancellation deadline	Renewal priority	Is perpetual
-	Definitely renew	No

Internal contacts 3
Agreement lines 1
Usage data 1

Annual Reviews

Note
-

Statistics Preview

Cumulative access per time

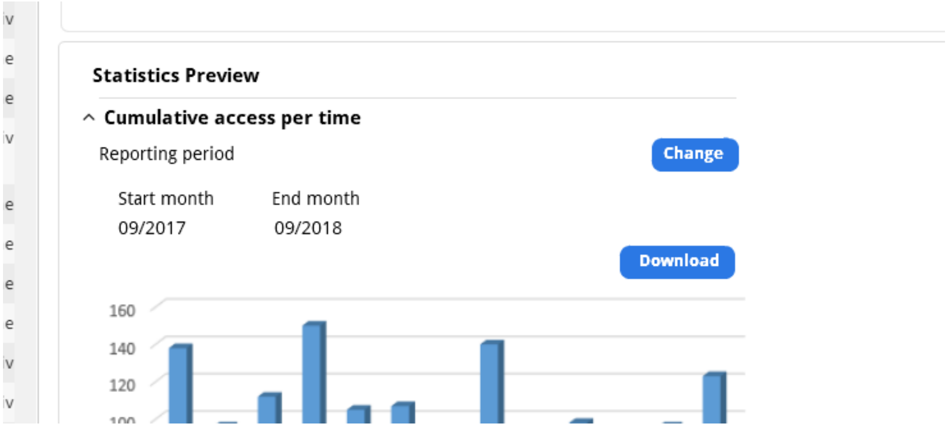
Reporting period

Start month	End month
09/2017	09/2018

Cost per download distribution across agreement
Access distribution across agreement
Percentage of low-level-use titles

(10-USER)

- User clicks on “Download” and gets an xlsx, ods or csv file consisting of both data table and visualization as well as metadata on agreement, reporting period etc.



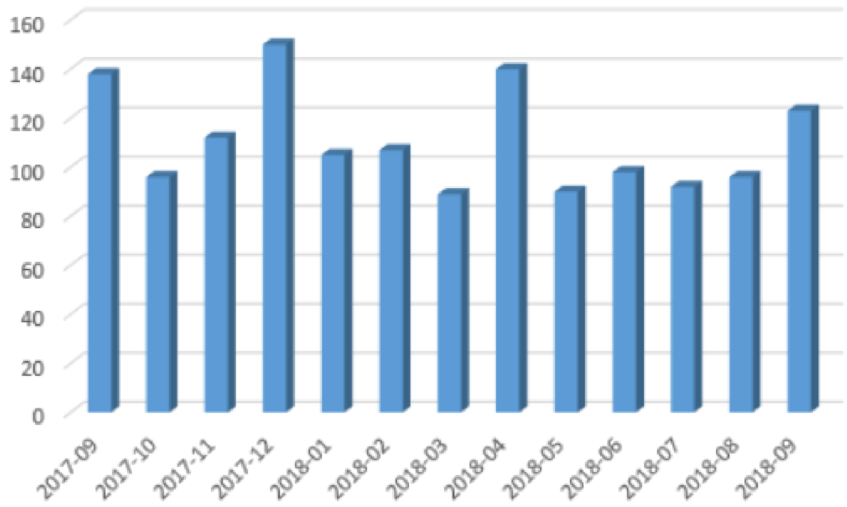
Statistics Preview

Cumulative access per time

Reporting period

Start month 09/2017

End month 09/2018

Statistics	Cumulative journal access per time
Agreement	Annual Reviews
uuid	1234-5678-9012-3456
Agreement period	2019-01-01 to 2019-12-31
Reporting period	2019-01 to 2019-06
Month	sum full text PDF
2017-09	138
2017-10	96
2017-11	112
2017-12	150
2018-01	105
…	
	

Further reporting scenarios to implement

The exact requirements for the reports to be developed are subject to a discussion in the ERM Sub SIG within the agile process of software development. The following report requests are known so far:

- Cost per download per agreement
- Price developments over several years per agreement
- Cumulative access per journal/title and time in an agreement
- Comparison of the use of the current year vs. archive years

A total of max. 10 reports are to be implemented.

Settings requirements

The reports listed in the "Statistics Preview" should be configurable tenant-based in the FOLIO-Setting-App from the available reports.

Non-functional requirements

The software development should take expandability and maintainability into account, since further reports will have to be created and integrated in the future and the COUNTER scheme may also change. These extensions should also be possible for third parties, e.g. for developers in libraries, without major hurdles.

A documentation of the functionality as well as the source code is expected. All guidelines provided by the FOLIO project must be followed. A documentation for the creation of further reports is also part of the development contract.

Milestones

Milestone	Means of verification
1. Backend module is completed in the main features	The backend module is demonstrated by means of a jointly determined report scenario and supports the limitation to a reporting period. Source code is available on https://github.com/folio-org
2. Enhanced features and number of reports increased	The limitation of report lines according to own entitlements list is demonstrated. (Only those journals remain in the report that are listed in the entitlements and therefore are currently subscribed). Two more jointly determined reports are demonstrated. Basic documentation is completed.
3. Completion of project; number of reports	Remaining reports are demonstrated. Documentation is completed.

increased	

Schedule, acceptance and payment

All deliverables are to be accomplished until March 31, 2021.

Partial invoices can be issued for completed milestones. For the examination and acceptance of the work packages, the contractor provides the SLUB Dresden with the documents and outcomes specified in the respective milestone descriptions (see section Milestones) by e-mail. SLUB as well as the Leipzig University Library is entitled to convene online meetings with the contractor if required. For this purpose, the library will contact the contractor within 2 weeks of submitting the results of a work package. The contractor must make an appointment possible that has been announced by the library 2 weeks in advance.

General conditions and constraint factors

All software components to be developed are subject to the terms of the Apache license, version 2.0. The source code has to be published at https://github.com/folio-org. The FOLIO project uses the agile management method SCRUM with biweekly sprints, sprint reviews and planning. Agile development includes agile and iterative requirements analysis. This means that the requirements specified in this RFP cannot be considered complete and elaborated. The further gathering of requirements is part of the project and is the responsibility of the client.

Neither SLUB nor Leipzig University Library can offer substantial technical support to achieve the project goal. In addition to a project start (kick-off) with detailed explanations and discussions on the scope of the project and the project goals, the client supports the contractor with regard to the availability of test data, the system configuration at user level and the test scenarios.

Both libraries cannot provide hardware, software, or software licenses. Travel expenses will not be reimbursed.