

MAAHANMUUTTOVIRASTO



MIGRATIONSVERKET



FINNISH IMMIGRATION SERVICE

INCREASING EFFICIENCY OF THE INSTITUTIONS OF THE INTERIOR SECTOR IN LATVIA

Phase 1

Recommendations for effective implementation of user workload accounts in information systems supporting business processes at the Office of Citizenship and Migration Affairs



Background information of the project

The **Office of Citizenship and Migration Affairs (OCMA)** currently faces problem with insufficient and ineffective control over the timing/controlling every-day-operational activities of its employee by making it impossible to efficiently analyse actual workload and appropriately delegate tasks. This situation produces risks to ensuring efficient organization of work processes as whole.

Analyses of the management processes performed by the audit company SIA Ernst & Young Baltic in 2015 underlined the necessity for establishing control system for measuring employee workload working with personal files ("user workload accounts") in the Unified Migration Information System.

OCMA has included the activity of creating "user workload accounts" in the Asylum, Migration and Integration Fund project, which is launched in 2018. Currently, OCMA seeks to obtain practical foreign experience for creation and implementation of "user workload accounts" to ensure the implementation of these accounts as successfully as possible.

OCMA user accounts - current situation

- Separate account and authorization for each IS;
- Task distribution is done in person or emails;
- No overview of the document circulation;
- Deadline control is done manually by employees;
- Manual statistics on business processes;
- No statistics about workload on employees and divisions;
- Many emails are not relevant for the recipient

OCMA user accounts - functionality

- Unified authorization (login/logout)
- Task distribution and management
- Deadline control
- Statuses
- Event notifications
- Statistics
- Activity log

OCMA user accounts

User

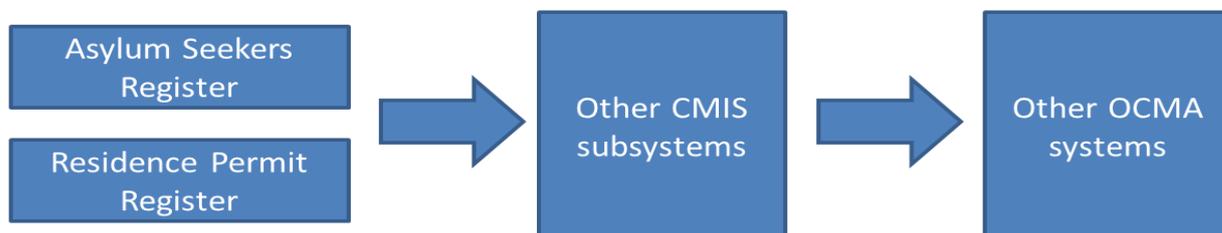
- Current tasks
- Notifications of current system events
- Scope of their rights
- Information systems to which he has access

Administrator

- Administrative tools
- Management
- Reports
- Activity log



Implementation of the OCMA user accounts



OCMA user accounts - questions

- Task distribution; how to distribute new cases, HR control/overview.
- Electronic document workflow; where is the case at this moment.
- Task fragmentation; approving applications etc.
- Deadline control; deadline for case, deadline for residence permit, deadline for different people in same case.
- User workload analytics; harder cases take more time, how to measure workload.

During meetings and workshops in September and October 2018 a lot of the topics of the scope “User accounts” have been discussed, demoed and presented to OCMA. These topics include for example:

- General information about Finnish Immigration Service (Migri), immigration in Finland, UMA IT-system and digitalization activities in the field of immigration
- User accounts in Migri’s UMA system
- Single-Sign-On and authorization services of UMA system
- Automation, rule engine and work management service in UMA system handling the electronic case processes
- Pre-requisites / pre-conditions and “checklists” in UMA system
- Actions and modules to handle data in UMA system
- Data exchange with different IT systems and data gathering
- Baskets (=configurable views of the cases / tasks / notifications) to replace static work queues in UMA system
- Tags to enrich the case / customer data
- Configurability of work queues (baskets), checklists, automation rules, task assignment, actions and modules
- Handling of electronic documents in UMA system
- Digitalization examples of residence permit and asylum processes
- UMA system navigation and layout
- UMA system high level technical architecture and services
- Customer electronic services - “EnterFinland” eService portal with recent customer feedback and “Kamu” chatbot



- Migri's customer service development program "Arvo2013"
- Organizational aspects of moving from papers to digital
- Change management and training aspects of moving from papers to digital
- UMA system delivery model (agile using scrum, SAFE, Kanban)
- Future digitalization plans including AI in different activities
- ResAid - appointment booking tool

Visits and workshops included visits to following locations:

- Population Register Centre (Helsinki); Population Registry tasks and activities and Suomi.fi services
- Migri customer service visiting point (Helsinki)
- Migri paper archives (Helsinki)
- Migri document scanning department (Helsinki)
- Accenture Riga Delivery Centre (Riga); Accenture Riga develops and maintains UMA system - their view on different relevant topics

All presentation and discussion material is delivered to OCMA and they can also be used to identify the appropriate recommendations.



Recommendations:

Recommendations are created as a result of the visits, workshops and discussion with OCMA. External opinions are also asked from residence permit and asylum experts in Migri. Accenture Riga Delivery Center senior experts also assisted in creating the recommendations.

- Single Sign On (SSO) service to full extent:
 - Since OCMA has a unified authorization service in registries under CMIS there is no urgent need to have SSO in the scope of project (residence permits and asylum). SSO can be part of the future phases of the project.
 - It is strongly recommended to have only strong authentication when introducing SSO. In the full extent there will also be IT systems owned by other organizations. Taking that into account the SSO should be provided by some organization / entity as a service for the whole government.
 - SSO and applying user rights digitally means that in CMIS there must be digital services for approvals / denials as well. Also expirations must be handled (including users leaving and changing their tasks)
 - Creation of SSO should reduce the work needed. While designing the services this must be evaluated.
 - Try to remove paper applications / forms of applying user rights as part of introducing SSO. Provide only digital services for applying user rights.
 - When only strong authentication is possible to establish, the weak (username, password) authentication should be removed. From security point of view the weakest point defines the level of security and parallel authentication methods also create extra work.
- User roles and showing the users the information about the scope of their rights (to different registries etc.)
 - Recommended to attach one user role to one user account. There might be reasons that multiple roles are needed.
 - The information about the scope of the rights should be easily understandable and the easiest way to create this is to have specific metadata describing the role.
 - Instead of very detailed user rights as part of the user role try to describe the rights in more general level (avoid fine tuning). With this approach the number of user roles will probably be lower and less testing effort and maintaining work of the roles is needed.
 - Separate the user roles by organizations / departments. When applying for the user rights it is simpler process to select the needed role(s). This also helps to move towards SSO services in the future and digital services to apply and approve new user rights or changed ones.
 - User rights could be implemented as a reverse process as well. Each role has high level user rights (keys) and then certain access is denied by the keys. Depending on the approach this might be considered as well.
 - Make sure that all needed user logs are available and perform log checks frequently and communicate that process clearly to all users.
- Work management: work queues and baskets (=views of the cases/tasks etc. meeting the configured criteria of a basket)
 - Implementing work queues is probably easier at first. It is recommended that configurable baskets will be experimented in parallel.
 - Implement work queues so that application/case is always in one work queue. It helps to make the change because there are no doubts that cases “get missing”. Migri hasn't experience of applications / cases being in multiple work queues at the same time - it might be possible but needs further studies.
 - Several levels of work queues are needed. From very general (agency and/or operational area) to single users (my work queue).



- If thinking about moving to assign tasks instead of assigning the whole application / case then tasks must be part of the work queues as well. Migri has no experience in this while using work queues and it might be difficult to implement that because of the primary nature of a work queue. Digital work queue is representing the paper work queue in digital environment. Assigning tasks might need other functionalities than just work queues.
- Replacing application driven case management with task driven case management has a huge efficiency potential but needs extensive training and organizational change management.
- Work queue rules must be generated and maintained; how applications / cases (or tasks) are moving. It is strongly recommended that some level of configurability is implemented because these rules will be changing by time.
- As part of the work queues there must be an easy way to transfer applications / cases from one work queue to another and select applications / cases to users' own work queues.
- Management needs similar services as part of assigning work to users. If a lot of assigning work is needed to multiple users at the same time (to help balancing the workloads) multiple work queues must be shown at the same time to help the assigning process.
- Integrating HR systems to IT systems might help identify the need to react to applications / cases that are in work queues of users that are absent. Depending on when the absence is entered in the HR systems. If selecting this option new technical services are needed to handle automatically the work queue transfers or to support manual ways of doing that. Easier approach in the beginning is to assign this kind of task to some specific people. Also alerts can be used to support to remind about applications / cases in the work queues of absent users.
- Configurable baskets provide full freedom of creating different "work queues". Using baskets the cases / applications are not attached to baskets in data model - baskets are views with of the cases that meet the criterias. Because of the benefits that baskets will provide it is recommended that technical capabilities and simple baskets will be experimented together with work queues.
- Baskets must be configurable by users (no technical work / project needed). It might be needed to limit the configurations.
- If baskets are used as part of the work management and as a work queues the primary (static) baskets are created by the persons that assign and manages the work of the users. These baskets can't be modified by all users.
- In baskets all varieties must be supported: applications/cases, tasks, alerts, notifications.
- Additional metadata ("tags") help to create the configurations of the baskets. Tags are understandable descriptions of for example application. "Tagging" can be both automated and manual. Configuration work must be practical - users doing configurations must not understand data models and database naming conventions which are usually (at least partly) technical.
- Baskets can be configured to support management / work load balancing need as well. This depends on the availability of structured data for this purpose.
- Every user should have a primary basket called "My basket" or "My tasks" when moving entirely to baskets.
- On top of configured baskets there must also be filtering, sorting and search functionalities into the baskets. This is especially important for the baskets that have big amount of content.
- Checklist, automation and rules with rule engine
 - As automation with rule engine is probably impossible to create in the first phase of the project it is recommended that checklists will anyway be implemented. Checklist describes all pre-requisites / pre-conditions / requirements that need to be completed to make a decision to an application / case. Checklist helps also to introduce the work to new employees.



- Checklists must have variants for each process / application profile. Also more general ones can be used if decided so.
- Each item in the checklist must show whether it is handled (decided) or not and also additional information / comments from the users (and later from the automation as well)
- Checklist must be configurable by users (those ones who have a right to configure them)
- Items in the checklist can be requirements from the legislation, practical formal requirements of an application etc.
- Checklist information can also be used as part of the basket configurations and task assignment process.
- Checklist describes the situation (status) of an application and it is recommended that the status of an application is recognized by the data (what has been done, what is missing / still needs to be done) instead of users setting the status / state / phase of an application manually. The status information is very important to users and processes. Therefore a lot of attention must be paid to describe the statuses as needed. Status information might be needed in other areas as well (eServices, transferring data to other systems, reporting and statistics (DW)) and that must be handled as part of the functional / technical design process.
- When moving towards rule based automations (checking the requirements automatically) checklists are anyway needed.
- Dashboards, reporting, user performance (activity)
 - Management needs dashboards to show the workload / backlog of their employees. Dashboards should be visual / graphical ones. Ideally dashboards could allow responsibility transfers from one user to another (workload balancing). In addition to single user dashboards also multiple user dashboards are needed to provide the needed services (workload balancing dealing with the bigger backlog)
 - Reporting can be used to look at the results / metrics of the employees performance.
 - As part of the user performance case complexity should somehow be taken into account. Case complexity evaluation is a difficult task to perform objectively. Initial case complexity estimations may change during the processing of an application / case. Recommendation is that case complexity is taken into account in application level first by describing self-obvious complexity differences. Complexity estimation is needed in the beginning of the process (workload balancing) and in the end (reporting the performance (activity) results of users.
 - Keep the complexity estimations transparent and let users participate in the estimation process.
 - Recommendation is that the scale to use for case complexity estimates is very high level (easy, medium, difficult).
 - When moving to task oriented case management the complexity evaluation becomes even more complex and might not be implementable objectively.
 - Complexity estimates can be used as part of the work queue and basket configurations. Complex cases can be assigned to more experienced users if so decided.
 - Supervisor <-> employee discussions are still needed as part of dealing with case complexities.
- Alerts and deadline control
 - If automation and rules engine is not implemented in the first phase of the project it is highly recommended that a simple rule engine to handle due dates and deadlines is introduced. Rule engine can run every night and set alerts to applications / tasks about deadlines.
 - Show the alerts in work queues, baskets and as part of the case information or have a specific UI for deadline alerts.
 - Simple rule engine for deadline control should create automatic alerts that originate from the legislation and common practices.



- Allow users to set manual alerts only for themselves or general ones for the application / case / task as an addition for the deadline control.
- As part of deadline control the defined alerts should be able to be modified or removed (modifying can be done by removing and creating a new one).
- Portal approach
 - Since CMIS registries are under the same authentication it doesn't create a need to handle authentication as part of the portal approach. However if taking into scope other registries / IT systems this must be handled.
 - There must be a unified data model / message description in each registry <-> portal interface to gather the needed work queue / basket etc. information to the portal. Optional data entities can be added to data models or message schemas for future / specific purposes.
 - Basket configuration on a portal level could become quite complex if one basket contains content from multiple registries. This must be evaluated if/when planning to use these kinds of baskets at portal level. Same applies for case complexity estimates and dashboards / reports / work balancing using this kind of horizontal case complexity information (comparing complexity between different case / application types).
- Electronic case management - documents
 - Some level of electronic case process must be defined for each case type / profile / processing ground. If/when using checklists it can ideally be high level as technical implementation - not as detail as normally is defined as part of the process work. Rule engine as part of the work management can control the case workflow - if it can't be implemented at first phase of the project the technical implementation should be as configurable as possible (practical configuration). Electronic document workflow / process is attached to electronic case workflow / process.
 - With the rule engine it is possible to create case workflow combining the automated tasks and manual tasks.
 - With the rule engine it is possible to trigger task assignments, notifications and alerts and add metadata <tags> (automatic tagging).
 - With the rule engine it is possible to create rules that end the case process before all pre-requisites are checked.
 - Moving from papers to electronic documents the best practices of paper world should be supported:
 - Opening multiple electronic documents at the same time (same case, different cases, comparing documents). Good practice for this is horizontal layout and two screens for each user (Note! one screen must still be supported)
 - Commenting documents, highlighting some information.
 - UI for all documents and preview of the first page. Marking documents as "important", "to courts", etc. and adding structured metadata.
 - Zoom, rotate, filter, sort, arrange.
 - In the future OCR possibilities could be used to identify documents automatically and extract information from the documents.
 - As digital documents contain information that is needed sometimes outside normal authorization process a temporary access to specific documents is a good practice in those situations. This can also be used for asking permission to electronically archived documents (simulating the use of paper document archive).
- Other recommendations
 - Instead of technical specifications or requirements it is better to describe practical requirements in tender. Technical specifications should only be used when there is an existing or pre-selected technical service / component / architecture that will be used. If detailed technical specification is described as a requirement it is the responsibility of the Customer (OCMA) to make sure that practical requirements can be fulfilled with the described technology / architecture.



- Moving from papers to digital needs strong investments in training and change management. There must be specific full time people responsible of these areas and a separate budget. If external organizations are also involved the change management process needs special attention to handle these organizations.
- Instead of big changes try to prove new ways of working with smaller pilot groups (and gather early feedback as well). Pilot - learn - improve - scale.
- Identify the areas where the biggest change is needed / will happen. Involve users as heavily as possible to participate in these areas. Use service design, early prototypes and user research / early user testing especially in these areas.
- Multi-year IT project most likely fail in terms of expectations if there is no ability to make the needed changes during the project as flexibly as possible. Usually these projects are fixed price projects and use waterfall project methods. Try to introduce agile methods if possible. One option would be that first part of the project is fixed price fixed scope project and others are good estimates and move towards agile project methods. Agile methods (scrum, safe, Kanban) need experienced people to make the change from traditional waterfall to agile.
- Configurability is initially an extra cost but it will be paid back when external services (development, technical configuration) are not needed to modify processes, checklists, work queues and their rules, baskets, rules etc. Configurability also allows experiments earlier because it is easier and faster to change certain things based on the feedback of the experiments. Extensive configurability might cause challenges to handle and maintain all configured items / areas.
- Structured good quality data is fundamental affecting all areas of digital case processing. Identify as early as possible the most important information needed. Also be aware that for example configurable baskets rely on data and incorrect data cause problems, lots of trouble-shooting and in worst case complex corrective actions.
- Data input is important. If possible identify the data input “modules” and implement these modules so that they are reusable.
- Simplify decision making if possible by having only one person doing the decision instead of two in the situations where it is possible (simple ones, positive decisions, etc.).
- Technical team (even if external company) should be involved as early as possible to understand the operations, requirements and targeted results / change. Early involvement increases the motivation and commitment.
- Electronic archiving is needed to fully get rid of paper driven case processing.