
Meta Resource Management System

Use Cases

René Freude (707168) <rene.freude@gmx.de>
Daniel Sadilek (707297) <sadilek@tfh-berlin.de>
Stephan Weiß (706830) <steph.weiss@web.de>

Copyright © 2003

2003-05-05

Revision History

2003-05-05

Initial public release.

Revision 0.1

This document specifies the use cases of the MRMS. Additionally it contains example uses of all use cases in one cohesive usage scenario and a glossary of the terms used.

Table of Contents

1. Usage scenario	1
2. Use Cases	6
2.1. User logs in	9
2.2. User logs out	10
2.3. User changes password	10
2.4. Create user account	11
2.5. Edit role assignment of user account	12
2.6. Create role	13
2.7. Set rights for role	13
2.8. Define resource type	14
2.9. Edit link rules	15
2.10. Create resource	15
2.11. Delete resource	16
2.12. Edit resource	17
2.13. Add resource link	17
2.14. Delete resource link	18
2.15. Relink resource	19
2.16. Create filtered collection of resource entries	19
Glossary	20

1. Usage scenario

This usage scenario presents an example environment for the MRMS. It serves as a concrete example from which we abstract the use cases. We have the problem that the MRMS is a very data centric system. So, usually you would begin to create the static analysis model and later derive the dynamic analysis model from it. Unfortunately, the time plan of the course, the MRMS is developed in, does not allow such an approach. Thus, this usage scenario serves also another purpose: Its lists and tables give you an idea of the data that will be managed by the system and helps you (and us) understanding the use cases better.

We assume that the system should be used in a business called Exemplar Inc. with approximately fifty employees. In this business there are to manage the employees, their workplaces with computers and tables, the rooms in which the workplaces are and the software licenses installed on the computers. An analysis of these resource types resulted in the following list of attributes that should be saved for the resources:

Table 1. Resource types at Exemplar Inc.

resource type	attributes
room	number, number of windows and maximum number of workplaces
workplace	description
computer	type of chassis and processor, processor frequency, size of memory and harddisk, monitor manufacturer and make
software license	name of software, type of license, license data, price, expiry date, maximum number of usages
table	material, size
employee	name, address, weekly hours of work, hourly rate

Resources of these types should be organized in the following manner:

- Rooms contain workplaces; the maximum number of workplaces for a specific room is saved as an attribute of the room.
- A workplace belongs to exactly one room.
- A workplace consists of at least one computer and one table but may also have more.
- A computer as well as a table belong to exactly one workplace.
- A computer may contain any number of software licenses.
- One software license may be installed on as many computers as saved in an attribute of the software license.
- Employees work at at least one workplace but may work on more.
- One workplace may be used by any number of employees.

Taking this information we can create a table of link rules:

Table 2. Link rules at Exemplar Inc.

resource type 1	cardinality 1	cardinality 2	resource type 2
room	1	0..room.maxWorkplace	workplace
workplace	1	1..*	computer
workplace	1	1..*	table
computer	0..swLicense.maxUsages	*	software license
employee	*	1..*	workplace

There are different roles with different responsibilities:

Table 3. Roles at Exemplar Inc.

role	responsibilities
chairman	manages employees and may also see and edit their hourly rate
server administrator	administrates the IT infrastructure in the server room and manages software licenses (e.g. buys new ones); is marked as MRMS administrator role and thus has administration access to the MRMS

role	responsibilities
client administrator	administrates and manages all computers that are not in the server room; assigns computers to workplaces and installs software licenses on the computers but may not manage the software licenses by themselves and has <i>no</i> administration access to the MRMS
facility manager	manages rooms, workplaces and tables: knows how many workplaces can be and how many are in a specific room and knows if a specific workplace is unused; assigns employees to workplaces but may not see the employees' hourly rate

The roles have different access rights that are assigned by an MRMS administrator, i.e. an owner of a role that is marked as MRMS administrator role (in our example this is the role “server administrator”). There are the following types of accesses (the points where the access rights are defined are in brackets):

- create a resource (resource type)
- delete a resource (resource type)
- read an attribute of a resource (attribute of the resource type)
- edit an attribute of a resource (attribute of the resource type)
- link two resources (link rule for the two resource types)
- unlink two resources (link rule for the two resource types)

The following table shows the access rights the roles in our example have. In the rows for the resource types “full” indicates that the role may create and delete resources of this type and may read and edit all attributes of resources of this type, “read” indicates that the role may read all attributes of a resource of this type. In the rows for the link types “full” indicates that the role may link and unlink the resources, “none” indicates that the role may neither link nor unlink the resources. Note, that reading/following a link is always possible for a role if at least one attribute of the link end can be read by that role.

Table 4. Access rights for the roles at Exemplar Inc.

resource type	chairman	server administrator	client administrator	facility manager
room	full	read	read	full
workplace	full	read	read	full
computer	full	read	full	read
software license	full	full	read	read
table	full	read	read	full
employee	full	read without “hourly rate”	read without “hourly rate”	read without “hourly rate”
link type	chairman	server administrator	client administrator	facility manager
room - workplace	full	none	none	full
workplace - computer	full	none	none	full
workplace - table	full	none	none	full
computer - swLicense	full	none	full	none

link type	chairman	server administrator	client administrator	facility manager
employee workplace	- full	none	none	full

The following employees work at Exempler Inc. and take one or more of the roles above:

Table 5. Employees at Exempler Inc.

employee	role(s)
Bernadette Bossa	chairman
Helene Hacker	server administrator
Chap Compscrew	server administrator and client administrator
Chris Clientow	client administrator
Florian Facilitan	facility manager

Installation

1. Helene Hacker installs the system on a server computer.
2. She logs in with the default administrator username and password; see Section 2.1, “User logs in” [9].
3. She changes the administrator password; see Section 2.3, “User changes password” [10].
4. She defines the types of the resources that should be managed (see in the table above). That includes specifying which attributes should be saved for a specific resource type; see Section 2.8, “Define resource type” [14].
5. She enters the link rules from the table above; see Section 2.9, “Edit link rules” [15].
6. She creates the roles of Exempler Inc. from the table above. This includes defining the rights that these roles have (see the table above); see Section 2.6, “Create role” [13] (includes Section 2.7, “Set rights for role” [13]).
7. She creates user accounts for all employees that take one of the roles; see Section 2.4, “Create user account” [11] (includes Section 2.5, “Edit role assignment of user account” [12]).
8. After these initial configuration steps she logs out; see Section 2.2, “User logs out” [10].
9. Then she informs the future users of the system about their usernames.

Initial data input

Now the system is set up and configured and the users have their usernames. They can begin to enter the resource data into the system; see Section 2.10, “Create resource” [15].

- Chris Clientow creates all computers he manages.
- Chap Compscrew creates all software licenses he manages.
- Florian Facilitan creates all rooms, workplaces and tables he manages.

(All of them log in for the first time and must set their passwords; see Section 2.4, “Create user account” [11].)

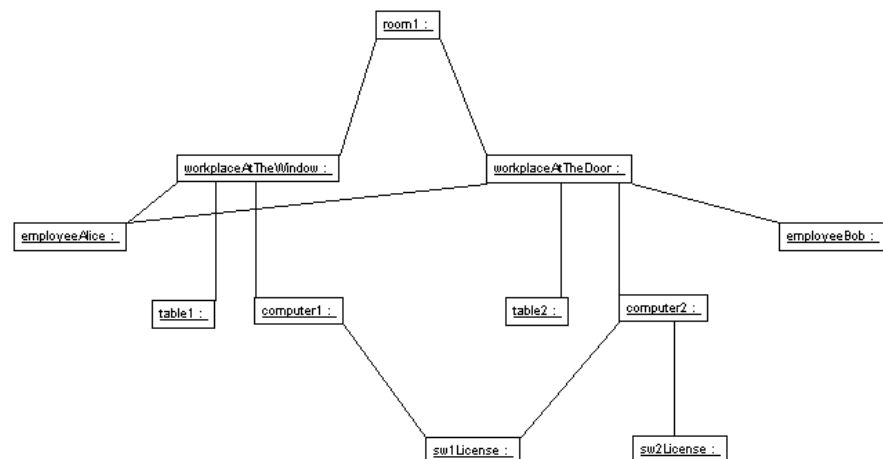
Initial link input

Now that all resources have been created in the system, the users create links between the resources to generate an image of the real resources' interrelationships; see Section 2.13, “Add resource link” [17].

- Chris Clientow links the software licenses to the computers they are used on.
- Florian Facilitan links the workplaces to the rooms they are in and to the computers and tables they consist of.

The following diagram shows a small excerpt from the data the users have entered. We will use it for the subsequent examples. You can see that there is just one room with two workplaces. Every workplace has its own table and computer. Both computers use the license for software 1 and computer2 (that belongs to the workplace at the door) also uses the license for software 2. Two employees work in this room: Alice and Bob. Bob uses the workplace at the door and Alice uses the workplace at the window. Sometimes, when she needs software 2, Alice works at the other workplace as well. (Note, that the “sometimes” is not saved in the system).

Figure 1. Some resources at Exemplar Inc.



Working with resources

Here we present some possible events that necessitate to work with the MRMS in order to correct the MRMS's image of the real resources or to find free resources:

Table 1 is old and broken and needs to be replaced:

1. Table 1 gets discarded.
2. Florian Facilitan deletes table 1 from the system; see Section 2.11, “Delete resource” [16].
3. If he would know check the consistency, he would be reported that the workplace at the window does not have any table which is required by the link rules; see Section 2.16, “Create filtered collection of resource entries” [19].
4. Another table, table 3, is bought and brought to the workplace.

5. Florian Facilitan creates table 3 in the system; see Section 2.10, “Create resource” [15].

Table 3 is not solid enough and needs to be replaced:

1. Florian Facilitan removes the link between the workplace at the window and table 3; Section 2.14, “Delete resource link” [18].
2. He does not create a new link with table 3, i.e. the table 3 now has not got any links to any other resources: it is a free resource. For this reason, Florian Facilitan transports the table to the business' storage.
3. Another table is bought, ...

Alice needs more memory for her computer 1:

1. Chris Clientow installs more memory into computer 1.
2. He edits the data of the resource “computer 1” in the system; Section 2.12, “Edit resource” [17].

The license for software 1 should be used on Alice's computer 1:

1. Chap Compscrew removes the license from computer 2 and installs it on computer 1.
2. He reproduces this in the system by relinking the license for software 1 from computer 2 to computer 1; see Section 2.15, “Relink resource” [19].

Bernadette Bossa needs an additional table:

1. Florian Facilitan searches for free tables in the system; see Section 2.16, “Create filtered collection of resource entries” [19].
2. He finds table 3 and transports it to Bernadette.
3. He reproduces this in the system by linking the table to Bernadette's workplace; see Section 2.13, “Add resource link” [17].

2. Use Cases

The following use case diagrams give you an overview of the use cases that are supported by the MRMS. Please note, that the administrator in the first diagram is also a user, but one with special access rights. So, actually, the administrator must have been connected to all use cases the user is connected to (for example “user logs in”), but in order to present a more concise overview we decided to connect him only to those use cases where he acts a special part.

Figure 2. User Management UCs

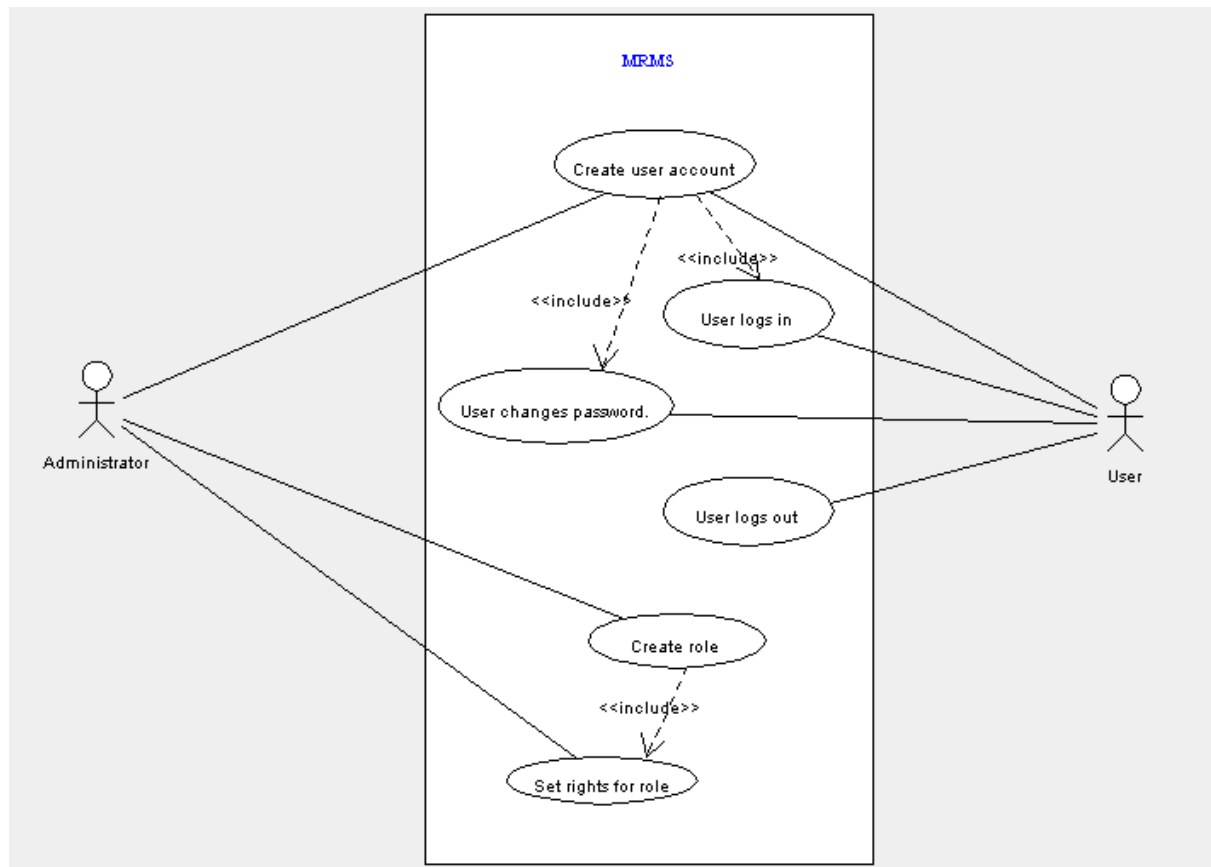


Figure 3. Meta Data Management UCs

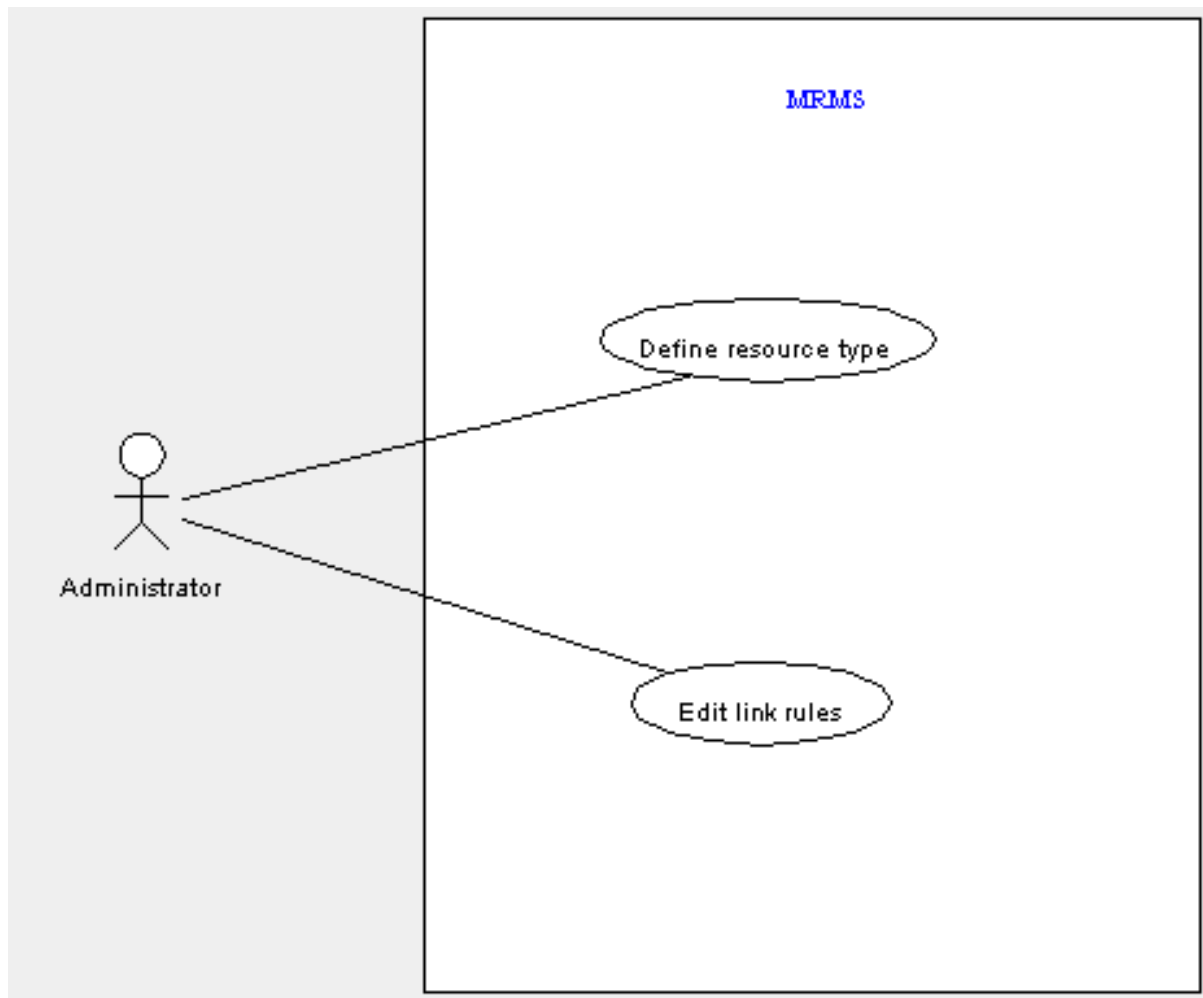
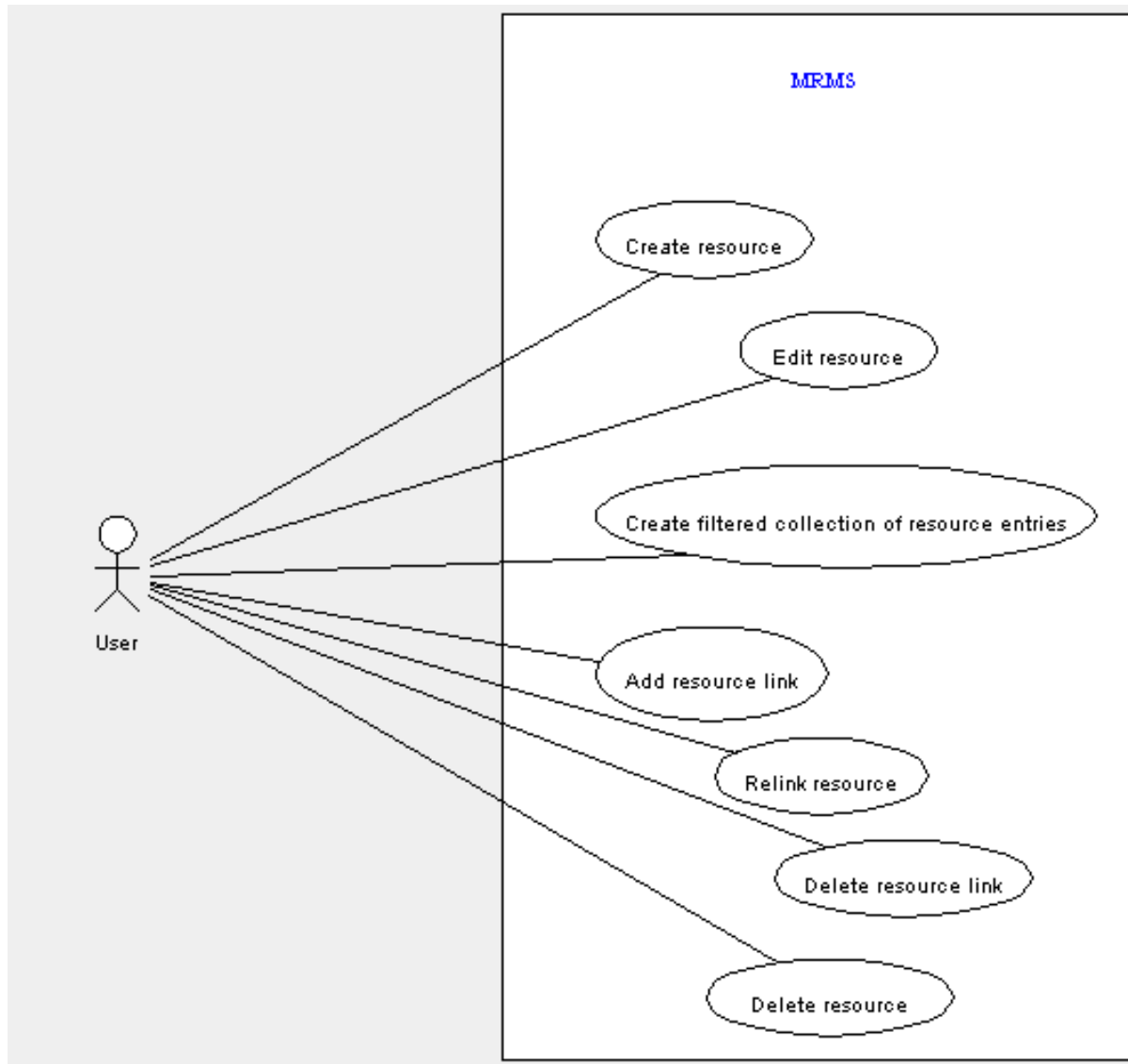


Figure 4. Data Management UCs



2.1. User logs in

Goal	The user logs into the system to be able to work with it.
Category	Primary
External Actors	User
Precondition	The user is not yet logged in.
Triggering Event	The user wants to work with the system.
Postcondition Success	The user is logged into the system and is able to work with it.
Postcondition Failure	The user must not be able to work with the system.
Description	1. The user starts/accesses the software. 2. A login input mask is opened by the system

3. He enters his username and his password.
4. He submits his input.

Extensions

- **4a.** If username and/or password are not valid the user is informed about this and prompted to correct it.

Alternatives ---

Additional Requirements ---

Annotation ---

2.2. User logs out

Goal	The user logs out from the system.
Category	Primary
External Actors	User
Precondition	The user is logged into the system.
Triggering Event	The user wants to end his working session.
Postcondition Success	The user is logged out and must not be able to work with the system without logging in again.
Postcondition Failure	The user remains logged in and is able to work with the system.
Description	1. The user requests the system to log out. 2. He is asked by the system to confirm. 3. The user confirms the logout action.
Extensions	---
Alternatives	---
Additional Requirements	---
Annotation	The user is logged out automatically after a specific inactivity timeout.

2.3. User changes password

Goal	The password for a user of the system should be changed.
Category	Primary
External Actors	User

Precondition	The user is logged into the system.
Triggering Event	The user (1) wants to set a new password for his account or (2) he must set it because the account has just been created and has still an empty password.
Postcondition Success	The new password has been set according to the user's input.
Postcondition Failure	The user's password has not been changed.
Description	1. The user requests the system to change his password. 2. A corresponding input mask is opened by the system. 3. The user enters his current password once and his new one twice. 4. The user submits his input.
Extensions	• 4a. If the current password is incorrect, the user is prompted to correct it. • 4b. If the two input values for the new password are not equal, the user is prompted to correct it. • 4c. If the new user enters a password that is too weak (less than six characters), he is prompted to enter a stronger password.
Alternatives	• 1a. If the account has an empty password, the input mask for changing the password is opened automatically without a user request.
Additional Requirements	---
Annotation	---

2.4. Create user account

Goal	Creation of a new user account.
Category	Primary
External Actors	Administrator, User
Precondition	A user is logged in who has administration access, i.e. who is an administrator.
Triggering Event	There is an employee who needs to work with the system and has not yet a user account.
Postcondition Success	A new user account with a non-empty password has been created according to the administrator's input.
Postcondition Failure	No new user account has been created.
Description	1. The administrator requests the system to create a new user account.

2. A corresponding input mask is opened by the system.
3. He enters the new user's username and his real name.
4. He submits his input.
5. The new user logs in with an empty password (likely via another client computer); see Section 2.1, "User logs in" [9].
6. He is automatically shown the change password input mask; see Section 2.3, "User changes password" [10].

Extensions ---

Alternatives

- **3a.** The administrator also enters a password. In this case steps 5 and 6 will not be executed.

Additional Requirements ---

Annotation Between steps 3 and 4 the administrator informs the new user about his username (and his password).

2.5. Edit role assignment of user account

Goal	The user's role assignment should be changed.
Category	Primary
External Actors	Administrator
Precondition	A user is logged in who has administration access, i.e. who is an administrator.
Triggering Event	(1) Because of a change in responsibility, a user needs to have other roles. (2) A new user account has been created which has not yet assigned any roles.
Postcondition Success	The assigned roles of a user have been changed according to the administrator's input.
Postcondition Failure	The assigned roles of a user have not been changed.
Description	1. The administrator requests the system to edit role assignment of a specific user account. 2. A corresponding input mask is opened by the system. 3. He selects or deselects entries from a collection of all possible roles. 4. He submits his input.
Extensions	---
Alternatives	---
Additional Requirements	---

Annotation	It is possible that a user account has not assigned any roles. In this case the user has no access rights.
------------	--

2.6. Create role

Goal	Creation of a new role in the system.
Category	Primary
External Actors	Administrator
Precondition	A user is logged in who has administration access, i.e. who is an administrator.
Triggering Event	A new role has been identified in the business for one of the following two reasons: (1) the existing access rights should be differentiated or (2) there is a new resource type which needs to be managed with indepent access rights from the already created roles.
Postcondition Success	A new role has been created according to the administrator's input.
Postcondition Failure	No new role has been created.
Description	1. The administator requests the system to create a new role. 2. A corresponding input mask is opened by the system. 3. He enters the role's name. 4. He submits his input. 5. He sets the access rights for this role; see Section 2.7, "Set rights for role" [13].
Extensions	---
Alternatives	---
Additional Requirements	---
Annotation	---

2.7. Set rights for role

Goal	Determining which rights users of a specific role have when working with the system and its data.
Category	Primary
External Actors	Administrator
Precondition	A user is logged in who has administration access, i.e. who is an administrator.
Triggering Event	(1) The rights for a specific role have to be changed because of changes in the business' set-up or (2) an initial setting for a newly created role has to be made.
Postcondition Success	The rights for the role have been set according to the administrator's input.

Postcondition Failure	The rights for the role have not been changed.
Description	1. The administrator requests the system to set rights for a specific role 2. A corresponding input mask is opened by the system. 3. For every resource type he selects if creation and/or deletion or none of them is allowed. 4. For every attribute of every resource type he selects if reading and/or writing or none of them is allowed. 5. For every link rule he selects if linking and/or unlinking or none of them is allowed. 6. He submits his input.
Extensions	---
Alternatives	---
Additional Requirements	---
Annotation	---

2.8. Define resource type

Goal	Defining of a new resource type.
Category	Primary
External Actors	Administrator
Precondition	A user is logged in who has administration access, i.e. who is an administrator.
Triggering Event	A new type of resources that should be managed by the system has been identified in the business.
Postcondition Success	A new resource type has been created according to the administrator's input.
Postcondition Failure	No new resource type has been created.
Description	1. The administrator requests the system to define a new resource type. 2. A corresponding input mask is opened by the system. 3. He enters the name for the resource type he wants to create. 4. He asks the system to create a new attribute for the resource type. 5. He selects the type of the attribute, if it's mandatory or optional and enters its name. 6. He submits his input. 7. If he wants to add another attribute he goes back to step 4.

8. He submits his input.

Extensions	---
Alternatives	---
Additional Requirements	--
Annotation	At least one attribute must be added to a resource type.

2.9. Edit link rules

Goal	Determine that two resource types can be linked together and specify the corresponding cardinalities.
Category	Primary
External Actors	Administrator
Precondition	A user is logged in who has administration access, i.e. who is an administrator.
Triggering Event	It has been identified that resources of two resource types must be linkable.
Postcondition Success	The link rule and its cardinalities have been defined according to the administrator's input.
Postcondition Failure	No new link rule has been created.
Description	1. The user requests the system to edit link rules. 2. A corresponding input mask is opened by the system. 3. He chooses two resource types out of a collection of available resource types. 4. He enters the cardinalities. 5. He submits his input.
Extensions	---
Alternatives	---
Additional Requirements	---
Annotation	---

2.10. Create resource

Goal	Creating a new instance of a resource type.
Category	Primary
External Actors	User
Precondition	A user is logged in who has proper access rights to create the new resource.

Triggering Event	A new resource should be managed by the system.
Postcondition Success	A new resource has been created according to the users input.
Postcondition Failure	No new resource has been created.
Description	1. The user requests the system to create a new resource. 2. A corresponding input mask is opened by the system. 3. He selects the resource type for the new resource. 4. He enters values for all attributes the resource type defines. 5. He submits his input.
Extensions	• 5a. If the input mask does not contain values for all mandatory attributes, the user is prompted to correct it.
Alternatives	---
Additional Requirements	---
Annotation	---

2.11. Delete resource

Goal	A resource that is managed by the system should be deleted from it.
Category	Primary
External Actors	User
Precondition	A user is logged in who has proper access rights to delete the resource.
Triggering Event	Either (1) the resource does not exist in the business any longer or (2) it should not be managed by the MRMS any longer.
Postcondition Success	The resource has been deleted from the system.
Postcondition Failure	The resource still exists in the system and no changings have been made to its attributes.
Description	1. The user requests the system to delete a specific resource. 2. He is asked to confirm the deletion. 3. He submits the deletion of the resource.
Extensions	---
Alternatives	• 2a. If there are resources that are linked to the resource that should be deleted,

the system asks the user if they should be deleted as well or if just the link to them should be deleted. If he decides for deletion and these resources are linked as well he will be asked the same question again, and so forth.

Additional Requirements ---

Annotation ---

2.12. Edit resource

Goal A resource entry should be modified.

Category Primary

External Actors User

Precondition A user is logged in who has proper access rights for editing the resource.

Triggering Event Real world information about an resource has changed and must be updated in the system.

Postcondition Success The resource has been modified according to the user's input.

Postcondition Failure The resource has not been modified.

Description

1. The user requests the system to change the attributes of a resource.
2. A corresponding input mask is opened by the system.
3. He performs the changes on the resource's attributes.
4. He submits his input.

Extensions

- **4a.** If the input mask does not contain values for all mandatory attributes, the user is prompted to correct it.

Alternatives ---

Additional Requirements ---

Annotation ---

2.13. Add resource link

Goal Creation of a new link from one resource to another.

Category Primary

External Actors User

Precondition A user is logged in who has proper access rights for adding the link.

Triggering Event	A resource's real world linkage has been extended and must be updated in the system.
Postcondition Success	A new link has been created between the specified resources.
Postcondition Failure	No resource link has been created between any resources.
Description	1. The user selects one resource as “link source” and requests the system to add a link. 2. The system displays a collection of possible resource types for the “link target”. 3. The user selects the resource type for the “link target”. 4. According to the selection, the system presents a collection of possible “link targets”. 5. The user selects one “link target”. 6. He submits his input.
Extensions	---
Alternatives	---
Additional Requirements	---
Annotation	The list of possible resource types for the “link target” in step 2 is determined according to the link rules: a resource type t1 will only be displayed if it appears together with the resource type of the “link source” in one of the link rules and if the “link source” has less links to t1-resources than the maximum cardinality allows.

2.14. Delete resource link

Goal	A link between two resources should be deleted.
Category	Primary
External Actors	User
Precondition	A user is logged in who has proper access rights for unlinking the resources.
Triggering Event	A resource's real world linkage is no longer existent and must be removed from the system.
Postcondition Success	The link has been deleted.
Postcondition Failure	The link has not been deleted.
Description	1. The user requests the system to delete a link. 2. The user is requested to confirm the deletion. 3. He confirms the deletion.

Extensions	---
Alternatives	---
Additional Requirements	---
Annotation	---

2.15. Relink resource

Goal	A link end r2 of a resource r1 should be substituted with another resource r3 (with same resource type as r2).
Category	Secondary
External Actors	User
Precondition	A user is logged in who has proper access rights for unlinking and linking the participating resource types; r1 and r2 must be linked.
Triggering Event	The real world linkage of resource r1 has changed and must be updated in the system.
Postcondition Success	R1 and r2 are not linked, r1 and r3 are linked.
Postcondition Failure	R1 and r2 are linked, r1 and r3 and not linked.
Description	1. The user selects the “link source” r1 and the “link target” r2 he wants to substitute. 2. The system displays a collection of possible alternative “link targets” of the same resource type as r2. 3. The user selects the resource r3. 4. The user submits his input.
Extensions	---
Alternatives	---
Additional Requirements	---
Annotation	---

2.16. Create filtered collection of resource entries

Goal	A collection of resources that pass a filter rule should be made available to the user for further processing.
Category	Secondary
External Actors	User
Precondition	A user is logged in.

Triggering Event	The user needs to find all resources meeting a specific criterion.
Postcondition Success	The user is shown a collection of resources that passed the filter rule he created.
Postcondition Failure	---
Description	1. The user requests the system to create a filtered collection of resources. 2. A corresponding input mask is opened by the system. 3. He selects the resource types to filter. 4. He selects the states he wants to pass the filter (consistent, linkable, free). 5. He submits his filter settings.
Extensions	• 3a. If he has selected only one resource type he may additionally enter filter rules regarding the attributes' values.
Alternatives	---
Additional Requirements	---
Annotation	---

Glossary

Free resource	A <i>resource</i> is called “free” if it has not got any links to any other resource; also called unlinked resource. Attend the difference to a <i>linkable resource</i> .
Link	The MRMS allows resources to be linked together according to the defined <i>link rules</i> . Every link is undirected and affects two resources.
Link end	If two resources are <i>linked</i> , they are called link ends.
Link rule	A link rule consists of two <i>resource types</i> whose instances - the <i>resources</i> - can be <i>linked</i> together. The link rule also specifies cardinalities which may either be fix numbers or attribute-values of the linked resources. Take the “room-workplace” link rule from Section 1, “Usage scenario” [1] as an example. Resource type 1 is “room”, cardinality 1 is “1”, resource type 2 is “workplace”, cardinality 2 is “0..room.maxWorkplace”. You can read this as follows: One workplace must be linked to exactly one room (cardinality 1); one room is linked to zero up to room.maxWorkplace (the value of the room's attribute maxWorkplace) workplaces (cardinality 2).
Link source	Although the <i>links</i> in the MRMS are undirected we use the terms link source and link target. When the user navigates along the links through the system, we call the resource he is currently using/editing/viewing the link source and all resources that are linked with this link source are the link targets.

Link target	See <i>link source</i> .
Linkable resource	A <i>resource</i> <i>r1</i> of <i>resource type</i> <i>t1</i> is called “linkable” in terms of another resource <i>r2</i> of resource type <i>t2</i> if <i>r1</i> has less <i>links</i> to <i>t2</i> -resources than the cardinality of the link rule for <i>t1-t2</i> allows. Attend the difference to a <i>free resource</i> .
Resource	The term resource can have two meanings: (1) a real object that is needed in a business to produce the business' products or services, e.g. computers, white boards, rooms, cars, etc. and (2) the virtual representation of a resource (in the first meaning) in the MRMS. A resource in the latter meaning has attributes that are defined by its <i>resource type</i>. In this sense a resource is an instance of a resource type. Resources can be <i>linked</i> together according to the <i>link rules</i> defined in the MRMS.
Unlinked resource	See <i>free resource</i> .