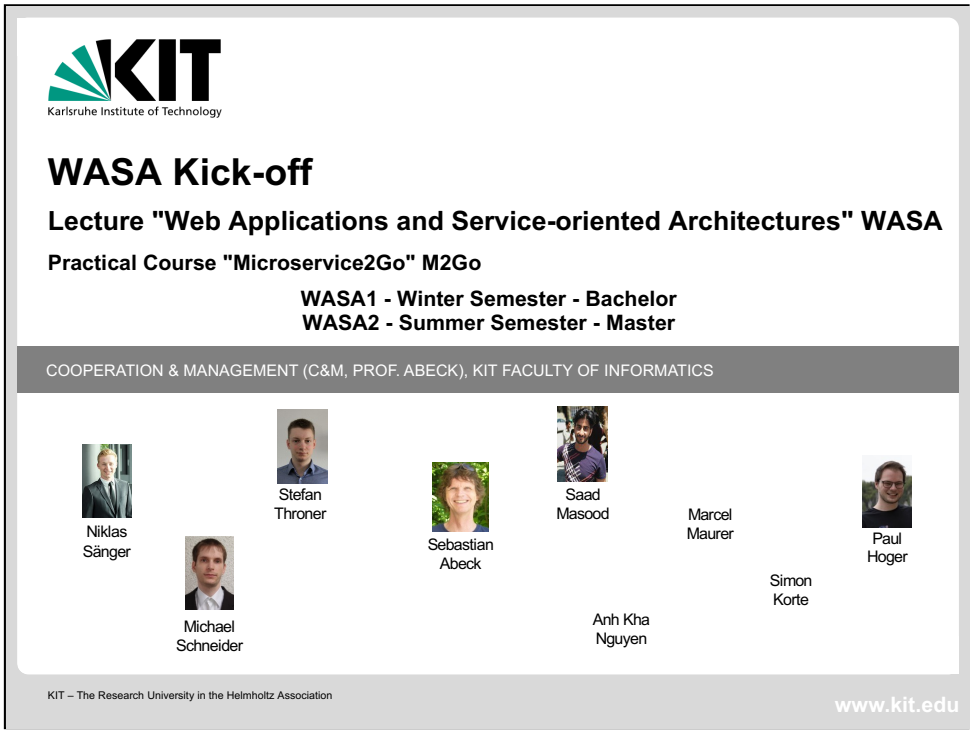


This course unit [CM-W-WAS] describes the content and the organization of the lecture "Web Applications and Service-oriented Architectures" (WASA) and the practical course Microservice2Go (M2Go) provided by the research group Cooperation & Management (C&M, Prof. Abeck). The M2Go practical course is closely linked with the WASA lecture. Although only the WASA lecture can be taken without passing the M2Go practical course, it is recommended to participate both in WASA and M2Go in parallel. Since the number of WASA lecture places and M2Go practical course places is limited, interested students must apply for a place.

WASA1 (Bachelor): Current concepts of software development and architectures (including Microservices, REST, gRPC, Domain-Driven Design, DevOps, CI/CD, Build Pipelines, Container-virtualized Infrastructures) as well as related technologies and tools (including HTTP, Go, Swagger, Postman, JavaScript/TypeScript, Angular, GitLab-CI, Docker, Kubernetes, Prometheus) are introduced. These concepts and technologies are applied in a systematic engineering approach, called Unified Microservice Engineering (UME), to develop and deploy microservice-based web applications.
The practical course Microservice2Go1 (M2Go1) can optionally be taken in parallel to the WASA1 lecture. In M2Go1, the UME approach is practically applied with the example of a microservice-based car rental application. After a compact introduction to the programming language Golang, the M2Go1 participants learn the systematic engineering of a domain microservice and an application microservice which are both implemented in Golang.

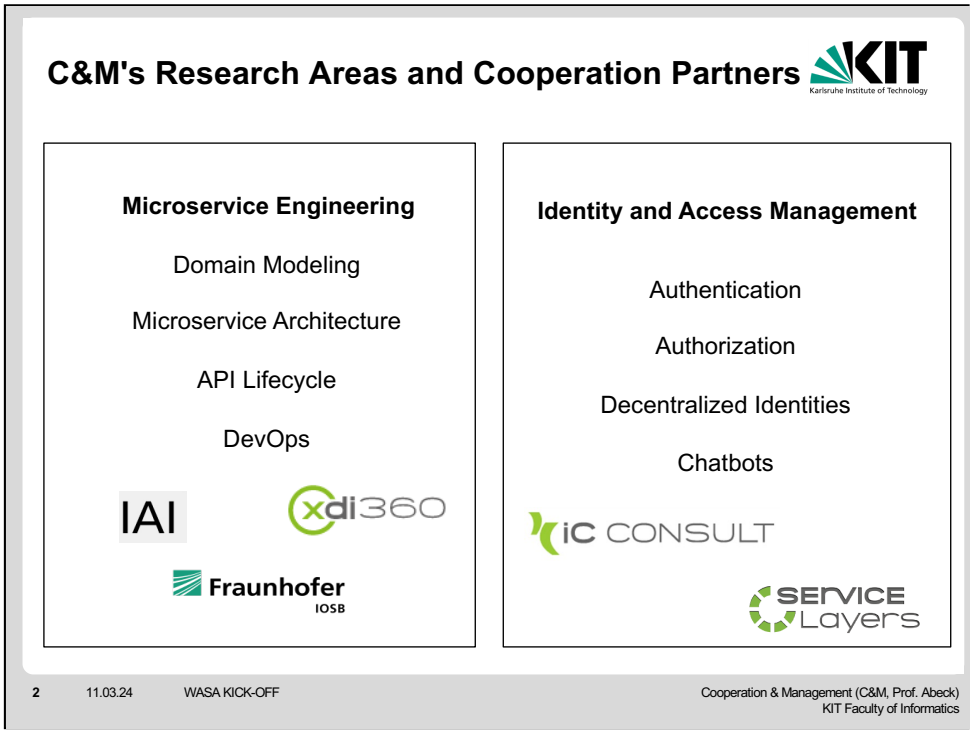
WASA2 (Master): A compact summary of the concepts covered by WASA1 is provided. In WASA2, Identity and Access Management (IAM) as an advanced topic is presented. In its core, IAM is responsible for the authentication and authorization of users and services in a software application. In the lecture, leading IAM concepts and solutions (e.g., Keycloak, Open Policy Agent, Microsoft Entra Verified ID) are introduced to illustrate how the IAM challenges are solved in IT practice. Concrete microservice-based applications are extended by authentication and authorization functionality based on the current Internet standards OpenID Connect and OAuth2.
The practical course Microservice2Go2 (M2Go2) can optionally be taken in parallel to the WASA2 lecture. In M2Go2, the IAM concepts presented in the lecture are practically applied on the existing analysis, design, and implementation artifacts of the microservices which are written in Golang.

The WASA lectures will be offered as a hybrid (i.e., mixture of presence and online) event. The WASA kick-off lecture will take place **online**
=== on Wednesday, 17th April 2024 at 9:45 am

Each student who wants to take part in the online WASA kickoff lecture must
=== send an email (in German) to cm.research@lists.kit.edu
to apply for one of the restricted WASA lecture (and practical) places. Please do **only** use your **depseudonymized** KIT student email address (see https://my.scc.kit.edu/shib/pseudonymisierung.php for further information).

The lecture material is made available in **English**. During the lecture, the content is presented and discussed in **German**. The **oral examination** is conducted exclusively in **German**. All students write their **practical/seminar thesis** in **English**. Thesis templates are made available in **LaTeX**. Overleaf is used for the writing of the practical and seminar thesis.

C&M	Cooperation & Management
IAM	Identity and Access Management
KIT	Karlsruhe Institute of Technology
M2Go	Microservice2Go
UME	Unified Microservice Engineering
WASA	Web Applications and Service-oriented Architectures

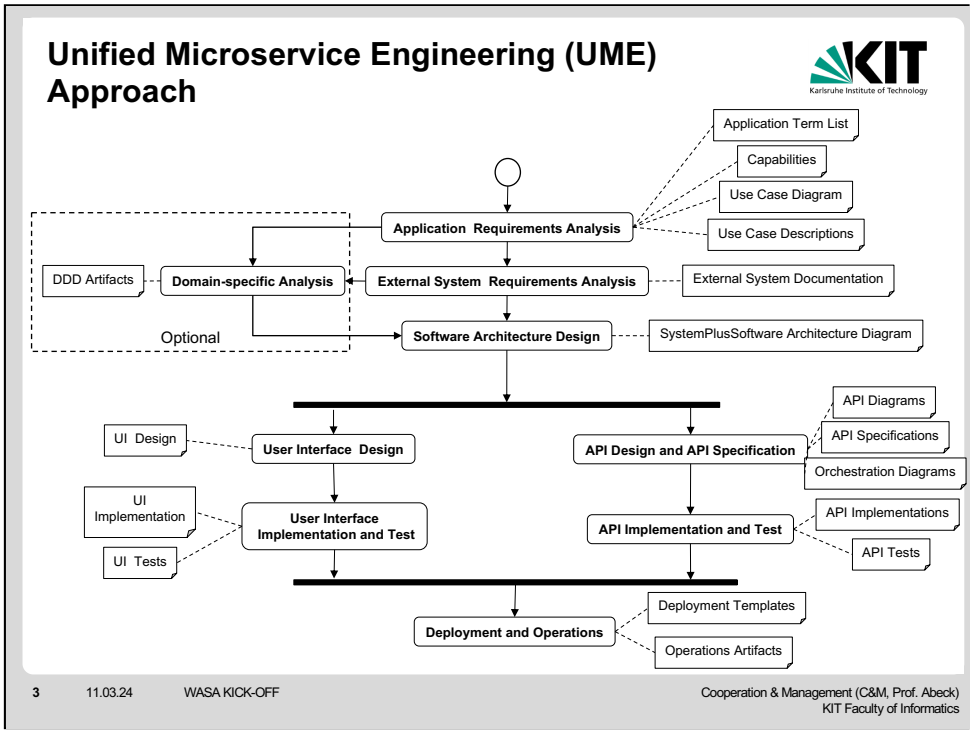


The research work carried out by C&M can be divided into two main areas:

(Microservice Engineering) For the business domain Connected Car, applications based on the concept of domain modeling and microservice architectures are developed. Relevant concepts applied in the microservice engineering approach include Domain-Driven Design (DDD), microservice API design and the implementation of the microservices and their APIs based on a microservice architecture. In addition to development (Dev), the operational aspects (Ops) are intensively taken into account.

(Identity and Access Management) Identity and Access Management (IAM) is a highly relevant crosscutting concern appearing in every web application. Basic IAM concepts include the authentication and authorization of human and technical users of a microservice-based application. Advanced IAM topics include
(i) Advanced authorization, which takes access decisions outside the application (external authorization) based on a broad spectrum of different attributes (fine-grained authorization).
(ii) Decentralized identities, which change the provision of identity in a way that users are owners of and have control over their identity data.
(iii) Usage of chatbot systems based on Large Language Models (LLM) to organize IAM knowledge.

CI/CD	Continuous Integration / Continuous Deployment
DDD	Domain-Driven Design
DevOps	Development and Operations
IAM	Identity and Access Management
LLM	Large Language Model



The Unified Microservice Engineering (UME) approach unifies two former approaches developed by C&M, called CMEng and MuleEng. UME consists of one process, the structured development process consisting of the well-known phases analysis, design, implementation and test, and deployment and operations. In the UME approach, the domain-driven aspects are not necessarily part of the engineering process (i.e., they are optional).

(Application Requirements Analysis) The requirements are expressed by use cases with a specific structure. Optional analysis artifacts are vision and goals or the application sketch.
(Application Term List) Application-specific terms are not introduced as a ubiquitous language. Ubiquitous languages are only provided by domains.

(External System Requirements Analysis) The external systems (esp. enterprise applications, business services, or databases) into the microservice application are considered.

(Domain-specific Analysis) In UME, the consideration of domain-specific aspects based on the concept of Domain-Driven Design (DDD) is intentionally kept optional. This analysis leads to Domain APIs which become part of the software architecture.
(DDD Artifacts) These include the ubiquitous language(s) and the domain model(s) of the domains relevant for the application to be developed.

(Software Architecture Design) The software architecture consists of (i) application microservices which are derived from the use cases, (ii) system microservices which integrate the external systems, (iii) optional domain microservices which provide the domain-specific logic, and (iv) Experience APIs which support the requirements of the different user interfaces.

(User Interface Design) (User Interface Implementation and Test) The User Interface (UI) can be designed and implemented and tested in parallel to the design and implementation of the different API types.

(API Design and API Specification) The API is systematically specified based on an API diagram. The API diagram of a Process API is derived from the use case descriptions and the software architecture diagram.

(API Implementation and Test) In UME, the microservices are coded and tested by using a Go-based framework.

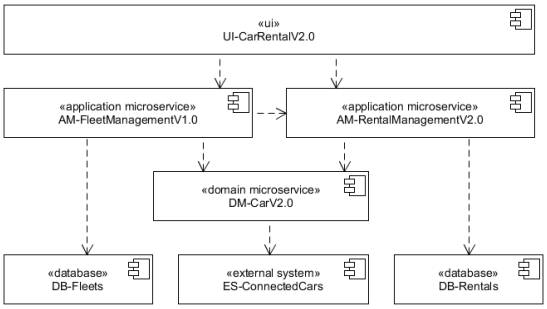
(Deployment and Operations) A template-based deployment approach is used in the UME approach.

UI	User Interface
UME	Unified Microservice Engineering

Component Diagram CarRentalAppV2.0

(1) CarRentalAppV2.0 includes two application microservices
  (1) AM-FleetManagement: Fleet manager can add and delete cars to / from a fleet
  (2) AM-RentalManagement: Customer can create and cancel rentals of cars

(2) AM-FleetManagement informs AM-RentalManagement about changes in the fleet
  (1) AM-RentalManagement is informed of all cars which are subject of a rental



4 11.03.24 AUTHN AND AUTHZ Cooperation & Management (C&M, Prof. Abeck) KIT Faculty of Informatics

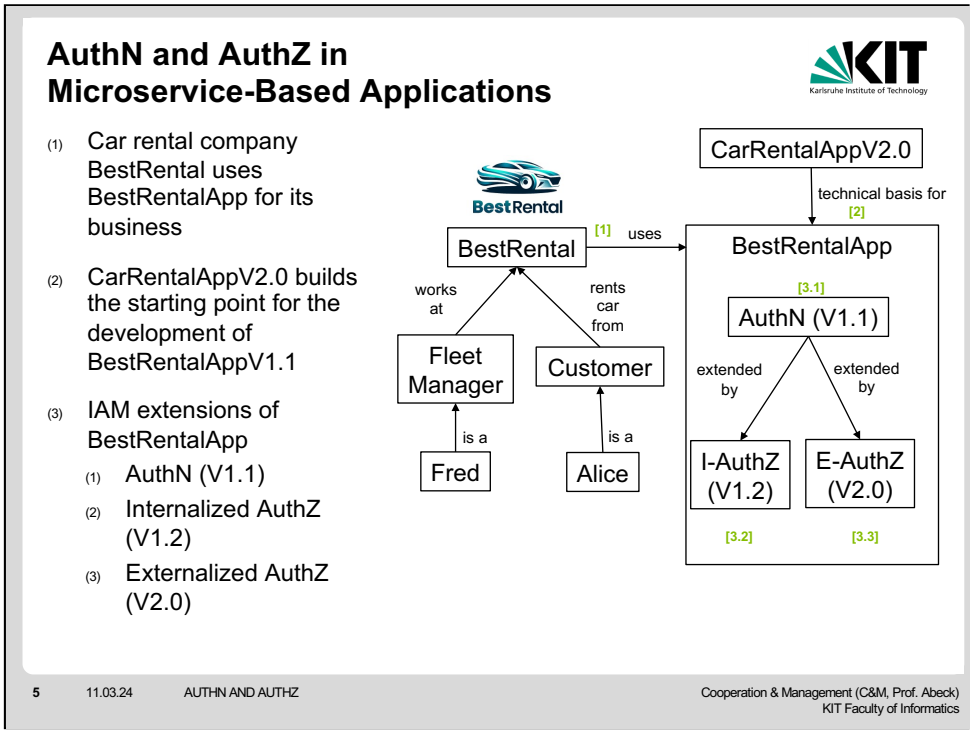
(1) The cars to be rented are organized in fleets. This leads to two different functionality parts to be provided by CarRentalAppV2.0, the management of the fleets and the management of the rentals.
(1.1) Initially only one fleet identified by its location is supported. For each fleet, one fleet manager is responsible. In CarRentalAppV2.0, no functionality to coordinate the fleets (e.g., creation of a new fleet, change of a fleet manager) exists.
(1.2) The functionality of AM-RentalManagement was implemented in CarRentalAppV1.0 (CarRentalAppV1.1 added the functionality of customer registration and deregistration to Version V1.0).

(2) The changes of the fleet especially concern the addition, replacement, and deletion of cars to and from the fleet.
(2.1) This means that AM-RentalManagement does not need to make requests to AM-FleetManagement to determine all available cars which is necessary to carry out a rental.

(«ui» UI-CarRentalV2.0) The user interacts with the UI-CarRentalAppV2.0 in order to rent cars and perform fleet management.

(«application microservice» AM-RentalManagementV2.0) The application logic provides the application-specific functionality in order to allow customers the rental of cars.
(«application microservice» AM-FleetManagementV1.0) The application logic provides the application-specific functionality in order to allow cars in a location to be organized in a fleet and to allow a fleet manager to manage his fleet.
(«domain microservice» DM-CarV2.0) DM-Car concerns the application agnostic functionality related to a car.

(«external system» ES-ConnectedCars) The external system from which DM-CarV2.0 retrieves its information about cars. It provides basic car information such as brand and model.
(«database» DB-Fleets) The fleet information is persisted in this database.
(«database» DB-Rentals) The rental information is persisted in this database.



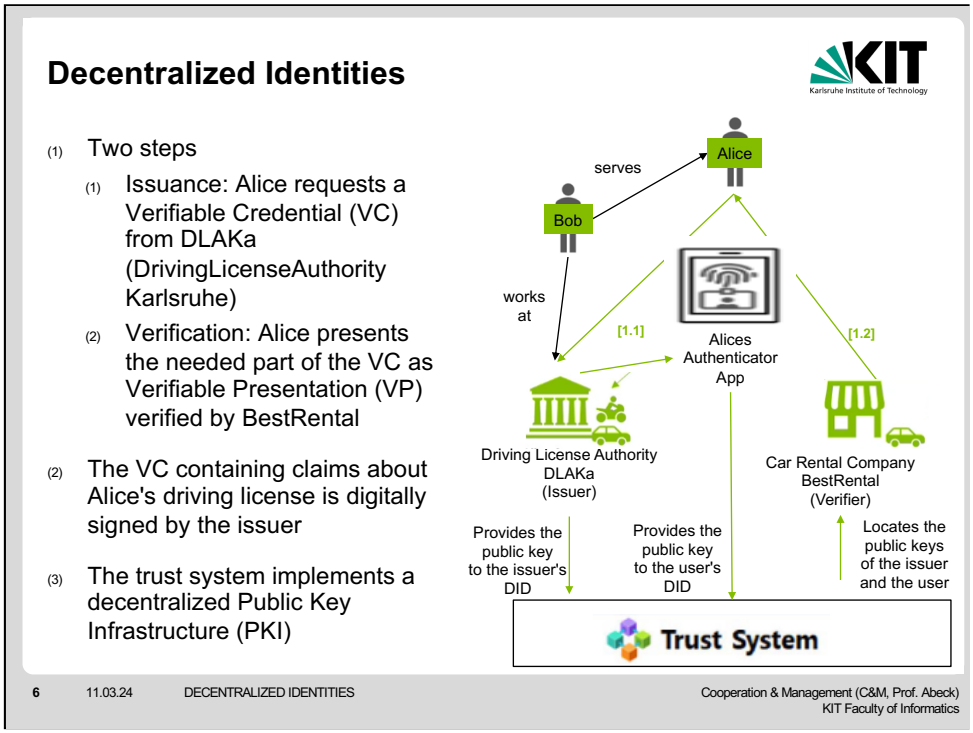
(1) For the demonstration of IAM aspects, a concrete organization called BestRental and a microservice-based application, called BestRentalApp, are introduced. Two roles, Fleet Manager and Customer are distinguished and two persons, Fred and Alice, are introduced.
(1.1) CarRentalAppV2.0 [CM-G-CRAV2.0] provides the basic functionality with respect to rental management and fleet management which is needed by BestRental. Therefore, CarRentalAppV2.0 builds the technical basis for the development of BestRentalAppV1.1[CM-G-BRAV1.1].
(BestRental) The logo is taken from a project work carried out at C&M.

(2) CarRentalAppV2.0 corresponds to BestRentalAppV1.0 which only virtually exists.

(3.1) CarRentalAppV2.0 will be extended by an AuthN (AutheNtication) solution which leads to BestRentalAppV1.1 [CM-G-BRAV1.1].
(3.2) BestRentalAppV1.1 will be extended by an Internalized AuthoriZation (I-AuthZ) solution which leads to BestRentalAppV1.2 [CM-G-BRAV1.2].
(3.3) BestRentalAppV1.1 will be extended by an Externalized AuthoriZation (E-AuthZ) solution which leads to BestRentalAppV2.0 [CM-G-BRAV2.0].

AuthN	AutheNtication
AuthZ	AuthoriZation
E-AuthZ	Externalized AuthZ
I-AuthZ	Internalized AuthZ

[CM-G-CRAV2.0] Cooperation & Management: CarRentalAppV2.0. https://gitlab.kit.edu/kit/cm/teaching/carrentalapp/carrentalappv2.0
[CM-G-BRAV1.1] Cooperation & Management: BestRentalAppV1.1, C&M GitLab. https://gitlab.kit.edu/kit/cm/teaching/bestrentalapp/bestrentalappv1.1
[CM-G-BRAV1.2] Cooperation & Management: BestRentalAppV1.2, C&M GitLab. https://gitlab.kit.edu/kit/cm/teaching/bestrentalapp/bestrentalappv1.2
[CM-G-BRAV2.0] Cooperation & Management: BestRentalAppV2.0, C&M GitLab. https://gitlab.kit.edu/kit/cm/teaching/bestrentalapp/bestrentalappv2.0



The proof of concept concerns the rental of a car at BestRental by the customer Alice. The approach of decentralized identities is used to proof to BestRental that Alice has a valid driving license. The scenario is adapted from [Mic-Dec] and [iC-LLD].

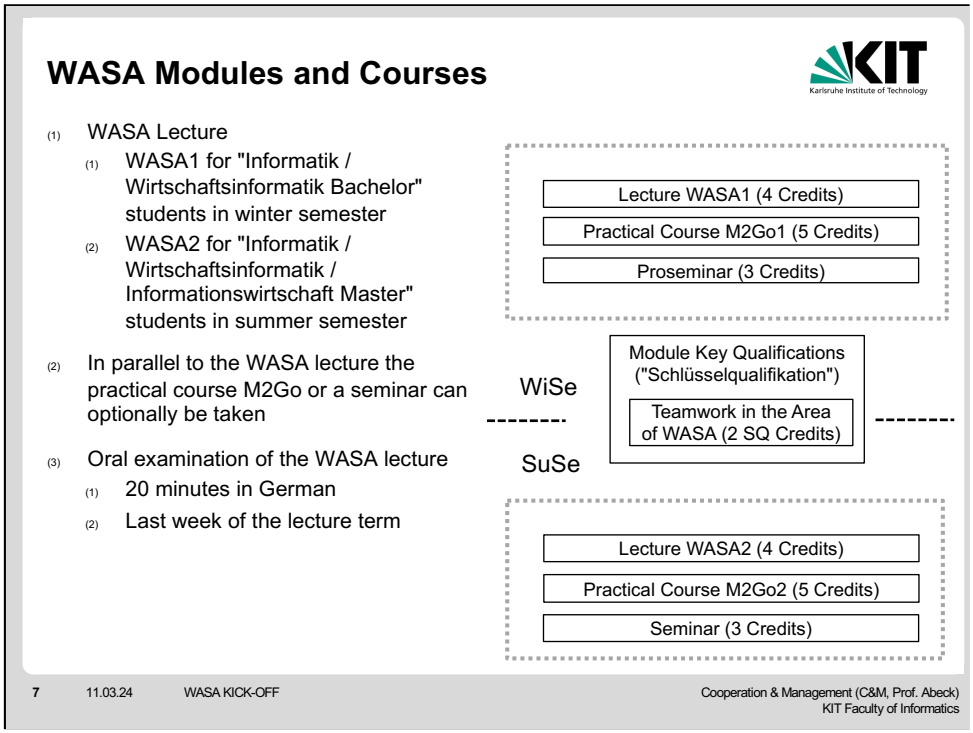
(1.1) Alice uses her wallet application to carry out the request to DLAKa, the Driving License Authority at Karlsruhe (Ka). She is served by Bob, who works as a DLA clerk at DLAKa. A signed Verifiable Credential (VC) is issued by DLAKa and stored in the digital wallet application, and which attests that Alice owns a valid driving license.
(1.2) Alice presents the VC on the BestRental website. The transaction is logged in Alice's wallet application.

(2) The claims contains attributes which specify (i) the type(s) of mobile vehicle, Alice is allowed to drive, (ii) the year when she passed her driving test.

(3) The VCs issued by DLA are digitally signed with the issuer's private key and the Verified Presentations VP presented by Alice (i.e., the user) to BestRental are digitally signed with Alice's private key. Thus, BestRental needs both the issuer's and Alice's public keys which are made available by a verifiable data registry. The trust system provides the public keys of the involved entities in a decentralized manner.

DLA	Driving License Authority
DLAKa	DrivingLicenseAuthorityKarlsruhe
DID	Decentralized IDentity
PKI	Public Key Infrastructure
PoC	Proof of Concept
VC	Verifiable Credential
VP	Verified Presentation

[iC-LL] iC Consult: Lunch & Learn: Decentralized Identities, restricted access. https://ic-consult.atlassian.net/wiki/spaces/EV/pages/3839033345/Lunch+Learn+Decentralized+Identities
[Mic-Dec] Microsoft: Decentralized Identity and Verifiable Credentials. https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE5cxkr?culture=en-us&country=us



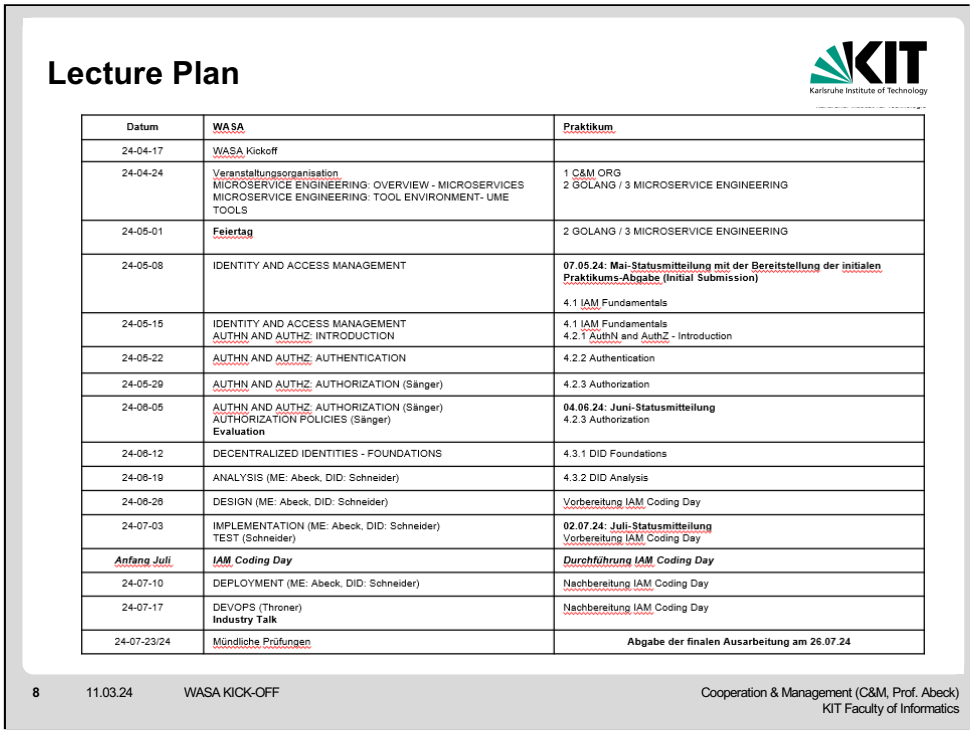
The acronym WASA stands for "Web Applications and Service-oriented Architectures". The following courses are offered: (i) lecture courses WASA1 an WASA2 (ii) practical courses M2Go1 and M2Go2 associated to the lecture courses (iii) proseminar course and seminar (iv) key qualification course (germ. Schlüsselqualifikation SQ).
Remarks: In the Wirtschaftsinformatik study programme the name of the module is "Microservice-basierte Web-Anwendungen".

(1) The lecture courses WASA1 and WASA2 each comprise 2 semester hours. A student who attends one of the lectures acquires 4 credit points (German: Leistungspunkt).

(2) The practical course M2Go is closely linked with the lecture course. The practical course counts 5 credit points meaning a workload of 150 hours. A proseminar and seminar count 3 credit points meaning a workload of 90 hours. A student who wants to carry out a (pro)seminar in parallel to WASA_M2Go will have a workload of 360 hours (4+5+3 = 12 credits) in the semester.

(4) The examiners are Prof. Abeck and one of the C&M's PhDResearchers. Since the examination is in the last week of the lecture term, the students should have a good personal resource management in order to have enough time for the preparation of the examination.

SQ Schlüsselqualifikation (Key Qualification)

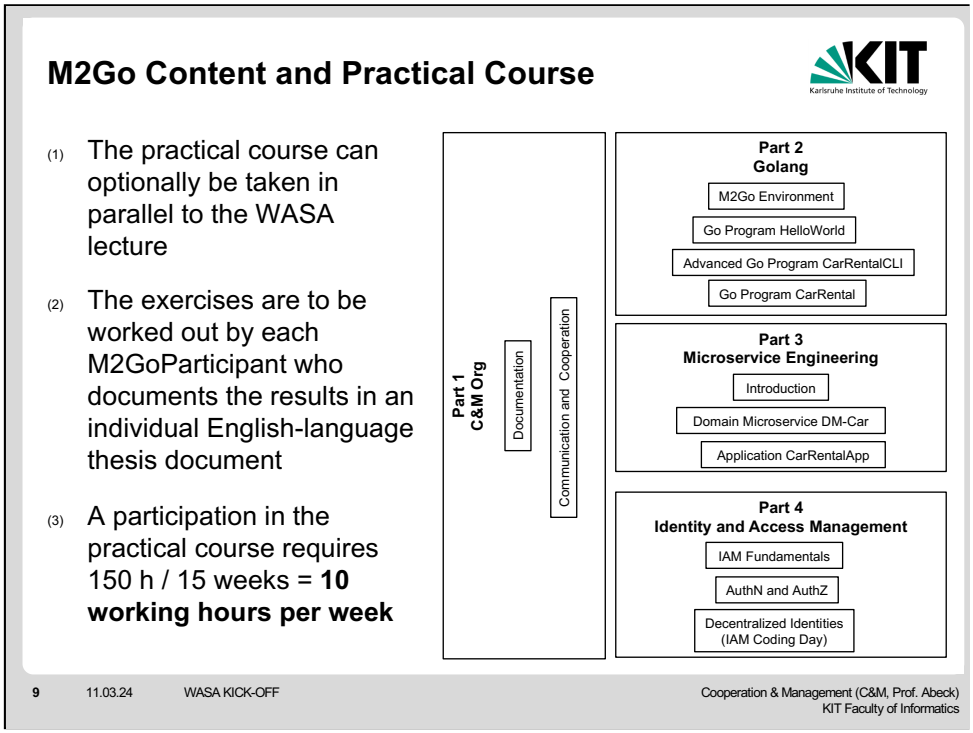


Lecture Plan

Datum	WASA	Praktikum
24-04-17	WASA Kickoff	
24-04-24	Veranstaltungsorganisation MICROSERVICE ENGINEERING: OVERVIEW - MICROSERVICES MICROSERVICE ENGINEERING: TOOL ENVIRONMENT- UME TOOLS	1 C&M ORG 2 GOLANG / 3 MICROSERVICE ENGINEERING
24-05-01	**Feiertag**	2 GOLANG / 3 MICROSERVICE ENGINEERING
24-05-08	IDENTITY AND ACCESS MANAGEMENT	**07.05.24: Mai-Statusmitteilung mit der Bereitstellung der initialen Praktikums-Abgabe (Initial Submission)** 4.1 IAM Fundamentals
24-05-15	IDENTITY AND ACCESS MANAGEMENT AUTHN AND AUTHZ: INTRODUCTION	4.1 IAM Fundamentals 4.2.1 AuthN and AuthZ - Introduction
24-05-22	AUTHN AND AUTHZ: AUTHENTICATION	4.2.2 Authentication
24-05-29	AUTHN AND AUTHZ: AUTHORIZATION (Sänger)	4.2.3 Authorization
24-06-05	AUTHN AND AUTHZ: AUTHORIZATION (Sänger) AUTHORIZATION POLICIES (Sänger) **Evaluation**	**04.06.24: Juni-Statusmitteilung** 4.2.3 Authorization
24-06-12	DECENTRALIZED IDENTITIES - FOUNDATIONS	4.3.1 DID Foundations
24-06-19	ANALYSIS (ME: Abeck, DID: Schneider)	4.3.2 DID Analysis
24-06-26	DESIGN (ME: Abeck, DID: Schneider)	Vorbereitung IAM Coding Day
24-07-03	IMPLEMENTATION (ME: Abeck, DID: Schneider) TEST (Schneider)	**02.07.24: Juli-Statusmitteilung** Vorbereitung IAM Coding Day
Anfang Juli	***IAM Coding Day***	***Durchführung IAM Coding Day***
24-07-10	DEPLOYMENT (ME: Abeck, DID: Schneider)	Nachbereitung IAM Coding Day
24-07-17	DEVOPS (Throner) **Industry Talk**	Nachbereitung IAM Coding Day
24-07-23/24	Mündliche Prüfungen	**Abgabe der finalen Ausarbeitung am 26.07.24**

Stored in C&M's document folder
Mitglieder > 2-1.WASA_M2Go_Aktuell
in the German-language PDF document
veranstaltungsorganisation_sose24

Remark: The lecture plan can only be accessed by students who take part in the WASA lecture.



(1) The WASA lecture content and the M2Go practical course content are closely coupled.

(2) For each M2Go part, exercise documents are made available. M2GoParticipants create their own practical thesis document in which they document the solutions of the exercises. The practical thesis document should make clear that an M2GoParticipant has carried out each part of an exercise. Therefore, the solutions should be long enough, but not longer. The text should be expressed in the own words of the M2GoParticipant. If text is copied, the source must be referenced. M2GoParticipants can make contributions to the presentation of the M2Go content in the WASA lecture. An English-language LaTeX document is made available in Overleaf.

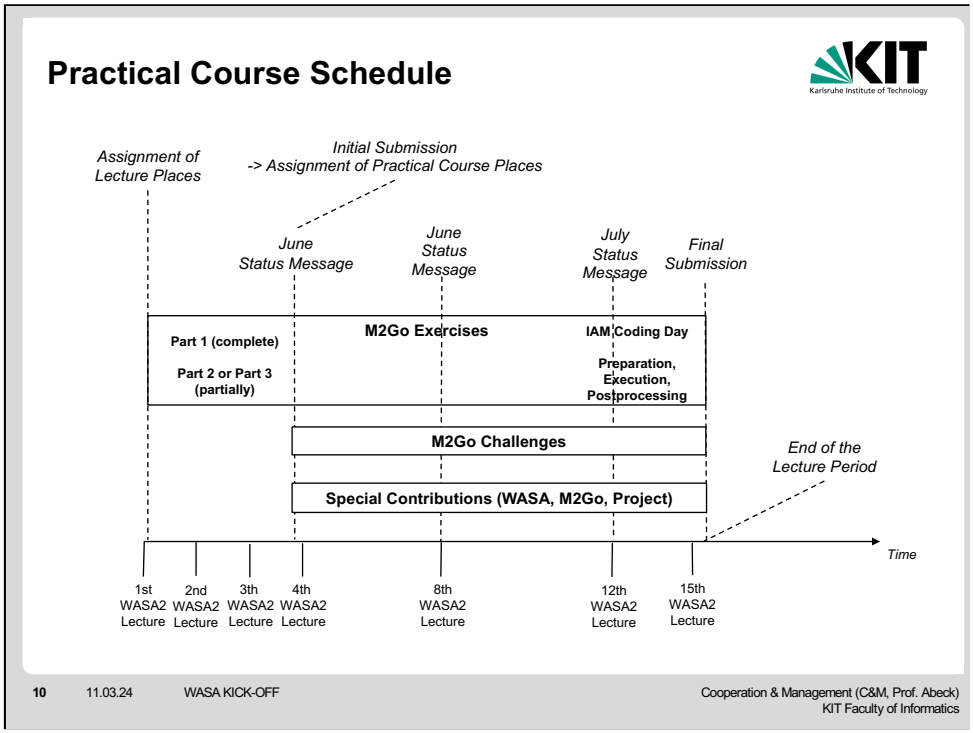
(3) The work starts immediately and the workload constantly arises every week. Therefore, students interested in participating in M2Go must make sure that they have enough time resources before they decide to do the M2Go practical course.

The practical course consists of the following four parts:

(C&M Org) All participants of the M2Go practical course must observe a few rules which exist at C&M to ease the cooperation between its members.
(Golang) This part provides a concise and practical introduction to the Go programming language. This includes the installation of the needed environment to code and run Go programs. Besides the well-known program "Hello World" some of the central Go language elements are investigated with the example of a more complex Go program CarRental and an advanced Go program CarRentalCLI.

(Microservice Engineering) In the following part, Go is used as the programming language to implement microservices of the application CarRentalApp. C&M's Unified Microservice Engineering (UME) approach introduces specific analysis and design artifacts which build the basis for the Go-based microservice implementation.

(Identity and Access Management) The last part is concerned with the relevant cross-cutting concern of Identity and Access Management (IAM). The access management includes the authentication and authorization of human and technical subjects who need access to the application BestRentalApp and their functions and data. A specific topic investigated in this part are IAM chatbots.



The figure illustrates the schedule of the M2Go2 practical course.

(Initial Submission) The practical course starts with M2Go exercises of Part 1 which makes sure that the WASA2Participant has successfully carried out the onboarding to the C&M environment and spends enough time for working out solutions of M2Go exercises.

(M2Go Challenges) The M2Go challenges are extensions of the M2Go exercises for which the M2GoParticipant works out individual and more complex solutions.

(Special Contributions) They go beyond the pure solutions of the M2Go exercises and challenges. Valuable special contributions are a prerequisite for a very good grade (i.e., 1,0 or 1,3).

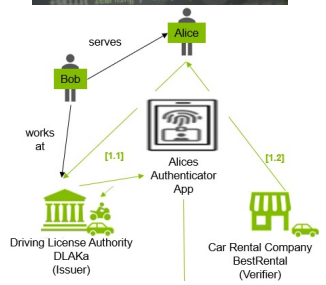
IAM Coding Day (ICD) 2024

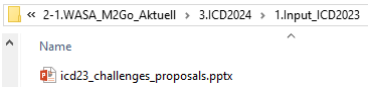
(1) ICD is a 24 hour hackaton offered each summer semester by iC Consult to WASA students
  (1) Mandatory part of the M2Go2 practical course

(2) This year's ICD topic: Decentralized Identities (DID)

(3) Business case: Issuance and verification of a digital driving license

(4) Challenges
  (1) Challenge collection of ICD2023 are a good starting point





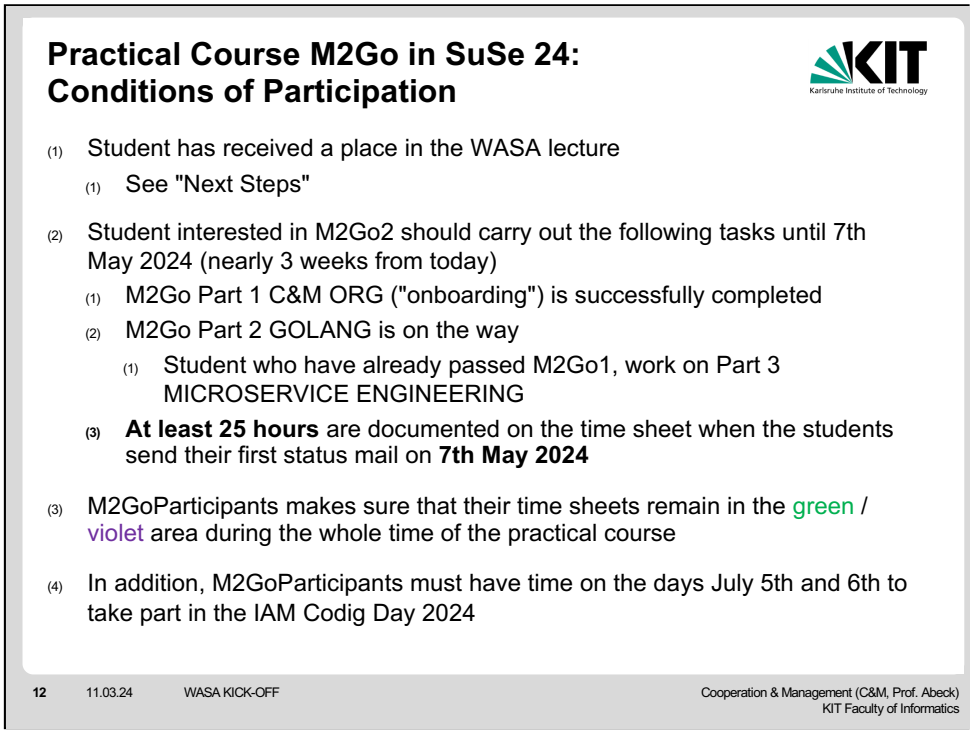
11 11.03.24 WASA KICK-OFF Cooperation & Management (C&M, Prof. Abeck) KIT Faculty of Informatics

(1) iC Consult is a leading consulting company in the area of Identity and Access Management (IAM). The IAM Coding Day (ICD) is an event which is an important element in the cooperation with the KIT research group Cooperation & Management (C&M).
(1.1) The knowledge transferred by the WASA lecture and the M2Go practical course is applied by the students to solve the ICD challenges.

(2) The concept of Decentralized Identities (DID) is conceptually and practically introduced in WASA and M2Go.

(3) The proposal is to extend C&M's driving license business case by two or three challenges.

(4) The challenges should cover aspects which are interesting for iCC.
(4.1) Challenges should be prepared in current Ba/Ma thesis (esp. Ba Maurer).

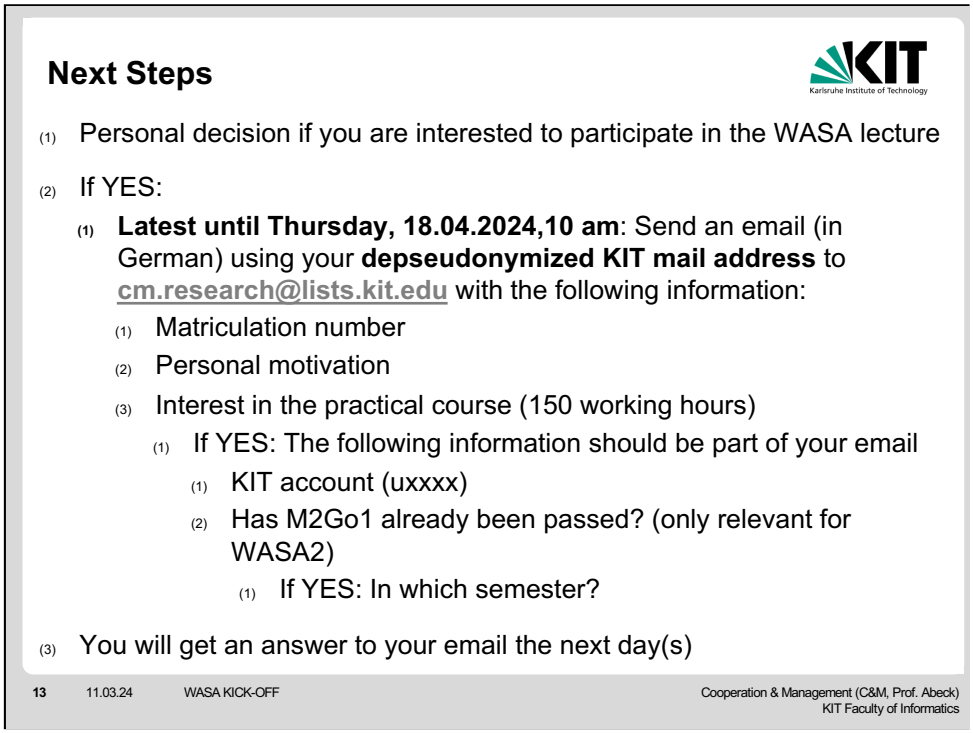


On this page the requirements are summarized which must be fulfilled to successfully participate in the M2Go practical course.

(1) A participation in the practical course M2Go2 is only possible in combination with the WASA2 lecture.

(2) It is explicitly noted that this is a condition which must be fulfilled by the interested student to take part in M2Go2.

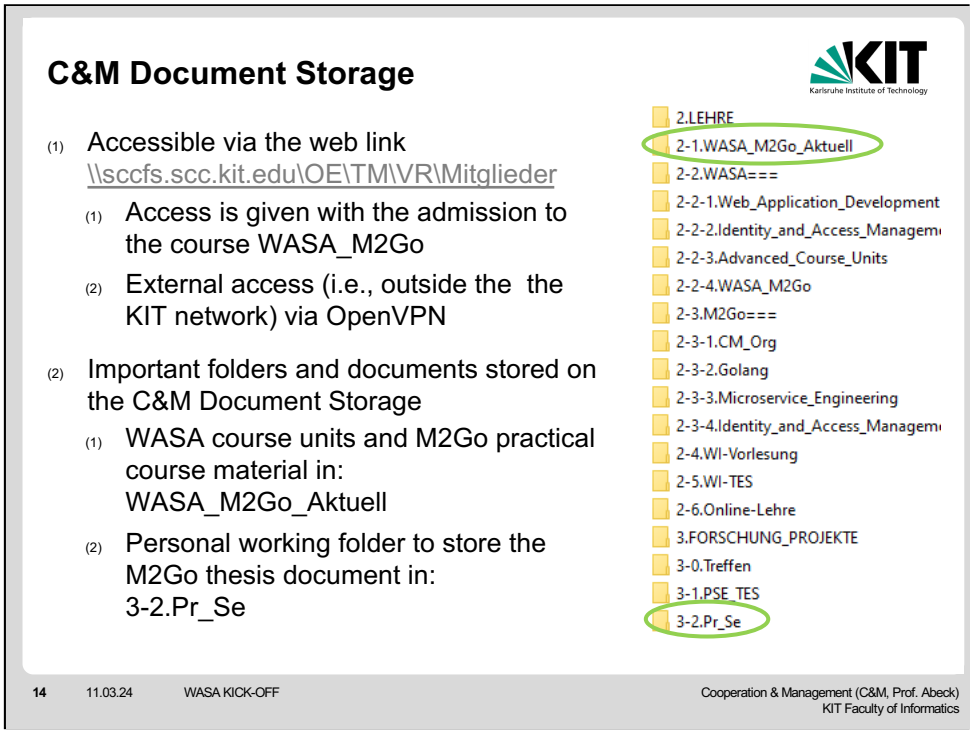
(3) This means that each M2Go participant should work about 10,7 hours per week on M2Go and document this work continuously in the time sheet.



(1) It is absolutely important that a student who participates in the WASA lecture and practical/seminar course has the necessary resources to cope with the workload (lecture: 120 hours, practical course: 150 hours, seminar course: 90 hours).

(2.1) Check on the page of the KIT Scientific Computing Center if your email is already depseudonymized.
(2.1.2) The motivation and the experiences should be summarized in at least one or two paragraphs.
(2.1.3) It is possible to apply only for the WASA lecture.
(2.1.3.1.1) The uxxxx student name is needed for the invitation of the M2Go participants to the C&M GitLab. An invitation requires that the M2Go participant has once logged in the GitLab.
(2.1.3.1.2) This part can be ignored in the case of WASA1.

(3) The email will contain the information if the student can take part in the WASA lecture. For those students who have has expressed their interest in the M2Go practical course, the email contains the information, if a practical and/or seminar place is assigned to the student.

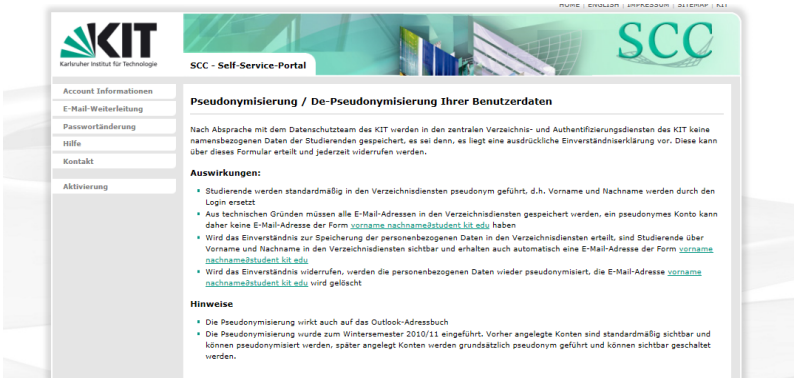


(1) Web link for Windows users: \\sccfs.scc.kit.edu\OE\TM\VR\Mitglieder
Web link for MAC/Linux users: smb://sccfs.scc.kit.edu/OE/TM/VR/Mitglieder

(2-1 WASA_M2Go_Aktuell) This is the most important folder which is available via \\sccfs.scc.kit.edu\OE\TM\VR\Mitglieder\2-1.WASA_M2Go_Aktuell

Activation of the Name-Related E-Mail Address (Depseudonymization)

(1) Can be carried out via Shibboleth (https://my.scc.kit.edu/shib/pseudonymisierung.php)



15 11.03.24 WASA KICK-OFF Cooperation & Management (C&M, Prof. Abeck) KIT Faculty of Informatics

This function can be found in the Studierendenportal by clicking on "Meine Benutzerdaten" > "De-/Pseudonymisierung" and accepting "Ich stimme der Sichtbarkeit meiner namensbezogenen Daten zu". Die name-related E-Mail-Adress <prename><surname>@student.kit.edu" exists additionally to the "uxxx@student.kit.edu" email address.