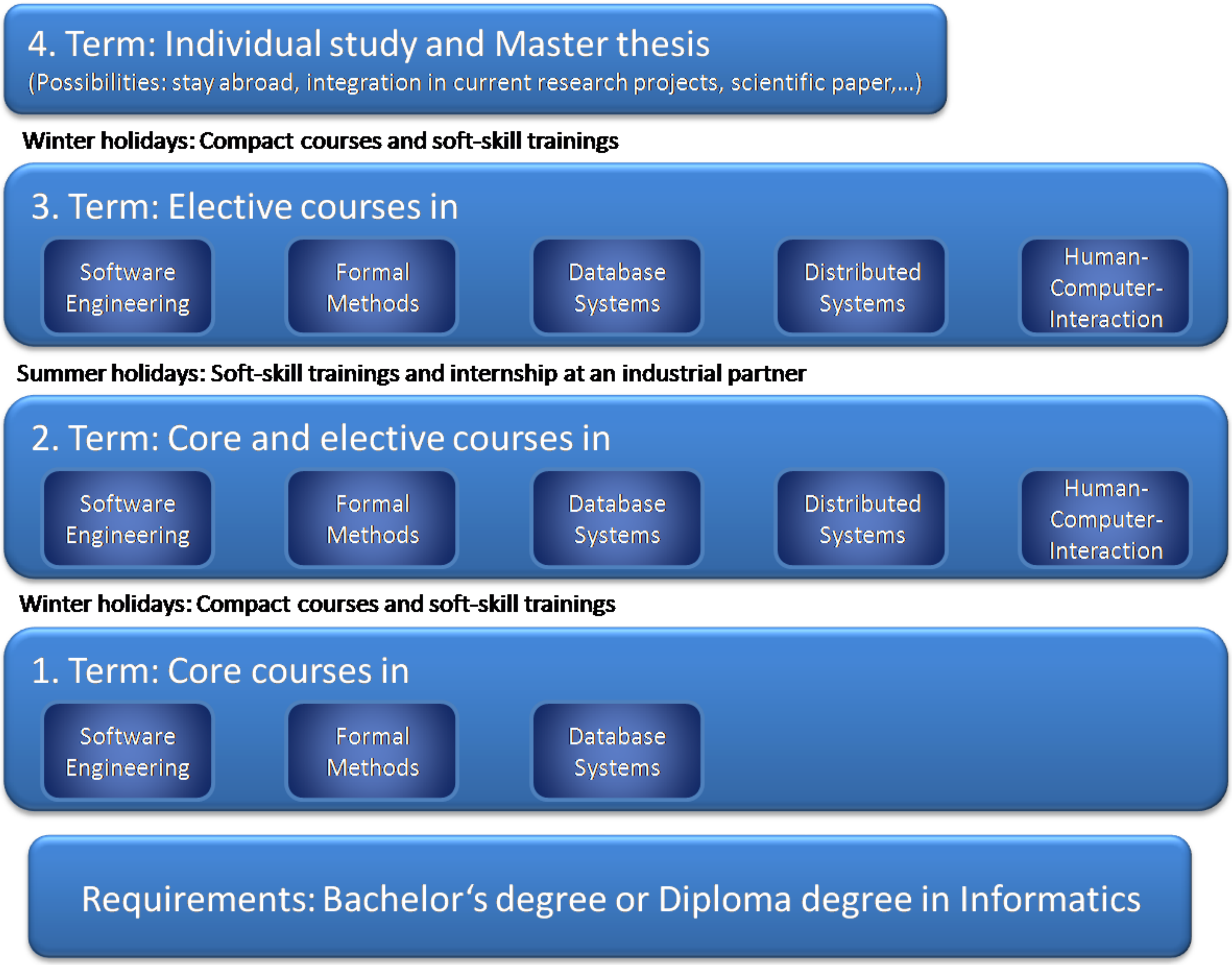
Software Engineering
Elite Graduate Program

Home
Application
Contact
Curriculum
Dates
Downloads
Faculty
FAQ
Links
Galleries
News
Partners
Our Students
Soft Skills
Mentoring

Curriculum

The course of study of the elite program ‘Software Engineering’ is organized as follows:



4. Term: Individual study and Master thesis
(Possibilities: stay abroad, integration in current research projects, scientific paper,...)

Winter holidays: Compact courses and soft-skill trainings

3. Term: Elective courses in
Software Engineering | Formal Methods | Database Systems | Distributed Systems | Human-Computer-Interaction

Summer holidays: Soft-skill trainings and internship at an industrial partner

2. Term: Core and elective courses in
Software Engineering | Formal Methods | Database Systems | Distributed Systems | Human-Computer-Interaction

Winter holidays: Compact courses and soft-skill trainings

1. Term: Core courses in
Software Engineering | Formal Methods | Database Systems

Requirements: Bachelor‘s degree or Diploma degree in Informatics

The education is focused on the following five central topics of modern **Software Engineering**:

- Software Engineering Methods
- Formal Methods and IT-Security
- Database Systems
- Distributed Systems
- Human Computer Interaction

The courses take place in Augsburg and Munich, with slightly more courses in Augsburg.

Integration of an individual study program

In arrangement with the Mentor a study program adjusted to the individual abilities and interests of the respective student is integrated in the education.The individual study program leads to the Master thesis and may include a stay abroad, research work or work in a project of an industrial partner. The individual study is planned for the fourth semester.

Practice orientation

The education in the planned course of studies is layed out in such a way that practice plays an important role. First of all the lectures are accompanied by exercises, which require the practical application of the concepts and techniques. Secondly the curriculum contains a practice project, which will be done by individual students or student teams in cooperating companies. Beyond that specialists from industry are merged directly into the teachings, in particular by guest lectures in the Elite colloqium as well as by the activity as a mentor. In addition direct contact with practice can also be part of the "Individual Study or Research" or an external master thesis.

Special lecture series

The special lecture series of the elite program Software Engineering takes place each semester during the teaching period. The special lecture series includes guest lectures from scientists from other research institutions to give the audience insight into current topics of research and new concepts in Software Engineering as well as talks from practitioners to complement the academic viewpoint with current trends and challenges from Software Engineering practice. Upcoming guest lectures are listed on the news page.

Courses

The following courses are regularly offered in the elite program:

First term

- **Software Engineering**: This course offers an introduction to current methods in systematic software development. Starting with the software life-cycle, the course deals with the different phases of software development and the currently relevant software development proesses. The main focus is the Unified Process and the usage of the UML in the different phases of development. Further topcis are testing, design patterns and GUI design.
  (6 credit points)
- **Formal Methods**: This course introduces the audience to techniques for the development of provably correct software.The course starts with techniques for an exact specification of the desired behavior and continues with a logic which can prove the correspondence between implementation and specification.
  (6 credit points)
- **Database Systems**: This course imparts knowledge on the design and operation of database management systems. Starting with the relational data model and its foundation, relational algebra, it continues with modeling and design techniques for databases, query languages, database access from OO-languages like Java and object-oriented databases. Furthermore it deals with transactions, multi-user synchronization, error handling, etc.
  (6 credit points)
- Compact course **Project-Management**: Project management is a key success factor for software development projects. Challenges for the project management are increasing productivity, ensuring quality, keeping the project within budget and timeframe. This course introduces the functionsor project management, relevant processes and methods and tools, from risk management to estimation methods.
  (6 credit points)

After the first term

Soft-Skills:

- Teamwork and conflicts in teams (2 credit points)
- Presentation techniques (2 credit points)
- Authors workshop (2 credit points)

Second term

- **Distributed Systems**: This course introduces fundamental concepts and methods for Distributed Systems development. Key topics are communication (Fundamental Intercation Model, Networks, Protocols and Services, Interprocess Communication), Processes, Synchronization and Coordination (Processes and Threads, Synchronization, Coordination) and Service-Oriented Architecture and Web Services (SOA, Web Services, REST, Web Service Composition).
  (6 credit points)
- **Human-Computer-Interaction**: This course introduces new paradigms for the interaction between humans and computers, the necessary theoretical background of modern interaction paradigms, specific methods for identification and interpretation of user input and for the generation and synchronization of system output. Another important topic are guidelines for usability in the design of interactive systems.
  (6 credit points)
- **Model-based Safety Analysis**: This course presents state-of-the-art techniques of safety analysis. Target of these techniques is the analysis of highly security-critical systems and they yield design guidelines to improve safety of these systems. Focus of this course is the increasing importance of software in embedded systems. The lecture starts with traditional safety analysis methods and then moves on to leading-edge techniques which are based on Formal Methods.
  (6 credit points)
- **Modelling, Specification and Verification of Reactive Systems**: Distinct aspects of reactive systems are concurrency, communication and non-termination. This course presents current approaches for modeling and specification of reactive systems and introduces refinement and verification techniques for reliable reactive systems.
  (6 credit points)
- **Requirements Engineering**: Requirememts Engineering is a crucial part of every development project. The gathered requirements are the basis of the development, integration and approval of the new system. This lecture starts with an introduction of the goals, responsibilities and topics of Requirements Engineering. It introduces different kinds or requirements, crucial steps in the process, methods and techniques of requirements gathering and the specification of the requirements. Further topics are structured documentation of requirements, use case and scenario modeling, non-functional requirements, requirements management and system models in Requirements Engineering.
  (6 credit points)
- **Testing**: Software Testing serves the evaluation and improvement of product quality by identifying problems and deficiencies. The course introduces the fundamentals of software testing, describes its position in the software development process and discusses test methods and tools as well as methods for test management.
  (5 credit points)

- **Model-based Software Engineering**: The goal of model-based software development is to describe (as far as possible) software systems through (semi-) formal models and to generate as many artifacts as possible from those models. This lecture gives an overview of the model-based software engineerin, focusing on the following main points: Modeling, meta-modeling, transformation and code generation.
  (5 credit points)
- **Database Systems on Modern CPU Architectures**:Implementation of Database Systems with respect to modern hardware: Storage Layer, Access Paths, Transactions, Set-Oriented Query Processing, Algebraic Operators. Understand the interaction between database systems resp. algorithms and modern computer architecture (esp. CPU, Cache, Primary Storage) and learn how to develop resp. to modify the internals of database systems in order to make use of the properties of these computer architecture features.
  (6 credit points)
- Soft-Skills
  - Leadership (3 credit points)

After the second term

- Internship at industrial partner (10 credit points)

Third term

- **Software Engineering for Distributed Systems**: Topic of this lecture are software architectures for distributed systems (software architectures and organizations, software architecture design, documentation of software architectures and evaluation of software architectures), semantic technologies (semantic web, ontology languages for the semantic web, description logics, reasoning with OWL and description logic reasoning) and multi-agent systems(intelligent agents, agent architectures, reactive and hybrid agents, approaches for decision finding, interaction in multi-agent systems and negotiation and coordination and cooperation through communication).
  (6 credit points)
- **Software Engineering for Embedded Systems**: Embedded systems are systems that are designed, integrated and operated in a technical environment for a specific purpose. Examples for fields of application of embedded systems are automobiles, airplanes, home appliances, mobile communication and consumer electronics. This lecture gives an overview on the design, the implementation and the technical environment of embedded systems.
  (6 credit points)
- **Practical IT-Security**: This course introduces fundamental concepts, techniques, methods and applications for IT security. Among other topics, the lecture covers threats and attacks on IT systems, goals of IT security, like integrity, secrecy and availability, cryptography, access control, authentication, network security and security engineering.
  (3 credit points)
- **Usability Engineering**: Because of the increasing pervasiveness of computers in all aspects of life and their embedding in the user’s natural environment, the shaping of the human-computer interaction is an increasingly important challenge. This lecture presents design principles and standards, the human information-, activity- and perception-processes, a user-centered development process and methods for the evaluation of interactive systems.
  (6 credit points)

- **Web-Databases**: In many cases, databases are accessible to the user via web-based interfaces. Examples are online shopping and web banking. The large number of users and the distribution of the application on different hard- and software systems like web servers, database servers, application servers, etc. led to the development of special software architectures and development paradigms. This lecture presents current programming languages, technologies and architectures for modern web-based information systems.
  (5 credit points)
- **Foundations of Data Engineering**: The course presents basic techniques for organizing and efficiently analyzing large amounts of data. The course discusses methods of distributed storage, distributed and parallel data processing, statistics-based optimization, map/reduce as well as different database implementations to support the analysis.
  (6 credit points)
- **Artificial Intelligence**: The course differentiates classical Artificial Intelligence from Machine Learning an discusses their application in the industry (e.g. Industry 4.0, Smart Factories, Smart Grids). The course presents different models and methods of AI, e.g. decision trees, regression, numerical optimization, and machine learning techniques.
  (5 credit points)

After the third term

- Soft-Skills:
  - Moderation techniques (2 credit points)
  - Intercultural competence (2 credit points)
  - Change management (2 credit points)

Fourth term

- Individual study in arrangement with the mentor, e.g.:
  - Stay abroad
  - Integration into the current research
  - Scientific paper
  - Internship The individual study leads to the
- Master thesis (30 credit points)

Home Application Contact Curriculum Dates Downloads Faculty FAQ Links Galleries News Partners Our Students