Transformation from OO to Microservices

Microservice architecture are replacing other software design and implementation paradigms. Previously adapted popular implementation paradigms of OO, took some years to develop and adapt teaching techniques preparing the new graduate for OO thinking. Today the main task in software solutions design is the breakdown of needed tasks into a very small fragment of microservice. Facing that challenges, a new teaching techniques and goals must be adapted. Assuming old ones will remain as the foundational skills needed from the software engineer, an exciting educational challenge emerges, and new teaching techniques/practices added to the software engineering curricula. We propose to use the old OO solution as the task enabler and microservice architecture as implementation strategy.

This paper is focused on the transformation done for the education of young software engineers attempting to use new teaching tools and approaches. We assume our students are already familiar with the basic coding skills and has experience in OO oriented software implementation.
This highlights work is the development and execution of new teaching techniques and tools.
Understanding that microservices are about functional decomposition often in a domain-driven design context. Microservices are characterized by well-defined and explicitly published interfaces. Each service is fully autonomous and full stack. Consequently, changing a service implementation should have no impact to other services, as communication takes place using interfaces only. Functional decomposition of an application and the new team view is the key to build a successful microservices architecture. Functional decomposition enables for instance agility, flexibility, and scalability. The following is an attempt to expose and motivate the students to a different design approach.

This work explains the different principles of model thinking and design. Relating it to the new reality needed in modeling the world. following by introducing Model driven design which vital for the analytical approach accompany Microservice architectural design.
Following, is a description of transformation process from monolithic solutions (implementation to a more granulate microservice architecture). This paper will demonstrate exercises and teaching technique in order to fulfil new way of observing the world which essential for microservice design and implementation.
We'll report and demonstrate this road thru student's assignment of redesign process.
Next, we propose exercises and training techniques for the young student in his journey for the ability to design microservice architecture driven application from scratch. Here are two examples of teaching program we have created for assisting this goal:

A. Student home assignment Monolithic to Microservice architecture design practice

Teams of a maximum of three students get a detail design document of a monolithic based application. And will have an assignment to redesign it using microservice architecture.

B. Experiencing Micro-Service buildup via LEGO bricks

Teaching micro-service architecture via exposing the student to LEGO® game building session, expose and motivate the students to a different design approach. The team are focusing to provide a simple but sophisticate model using the very basic articles of building stone – Lego breaks. Teams of a maximum of four students will be responsible for the design and building of a **vehicle**, consists of at least six independent components. The team are focusing to provide a simple but sophisticate model using the very basic articles of building stone – Lego breaks
Finally the paper will report statistics and information regarding outcome of these exercises analyzing the influence and impact on the students capabilities and achievements.