# Software engineering personnel training based on CDIO

**Nianqing Tang**

Neijiang Teachers College
Sichuan, People's Republic of China

ABSTRACT: In view of the traditional educational orientation towards disciplinary knowledge with weaknesses in innovation, practice and teamwork, software engineering personnel training based on CDIO (Conceive, Design, Implement, Operate) was implemented for the software engineering major of the Department of Computer Science at Neijiang Teachers College in Sichuan, China. The reform of the curriculum, teaching, practice environment, and guarantee and assessment systems are described in detail, to provide a reference for engineering education of the software engineering discipline.

## INTRODUCTION

National competition in the information industry is becoming increasingly fierce. As the key to the information industry and the foundation of informatisation of the economy, the software industry has received much attention in every country of the world. For the purpose of meeting the strategic adjustment of the Chinese economic structure and the urgent need of the software industry for skilled personnel, cultivating high-level interdisciplinary practical software talent having international competitiveness is an important task for China. How to foster new practical software talent is a significant research subject for software engineering.

The traditional software educational pattern is oriented towards disciplinary knowledge and is weak in developing innovation, practice and teamwork. Hence, the original educational pattern has been unable to meet the requirements of the software industry [1][2].

Taking the direction of the current software industry and the demands for software talent as a guide, by using advanced CDIO (Conceive, Design, Implement, Operate) engineering educational concepts and standards, the Department of Computer Science at Neijiang Teachers College has developed a model of software engineering personnel training, based on CDIO [3-6]. The model is shown in Figure 1.

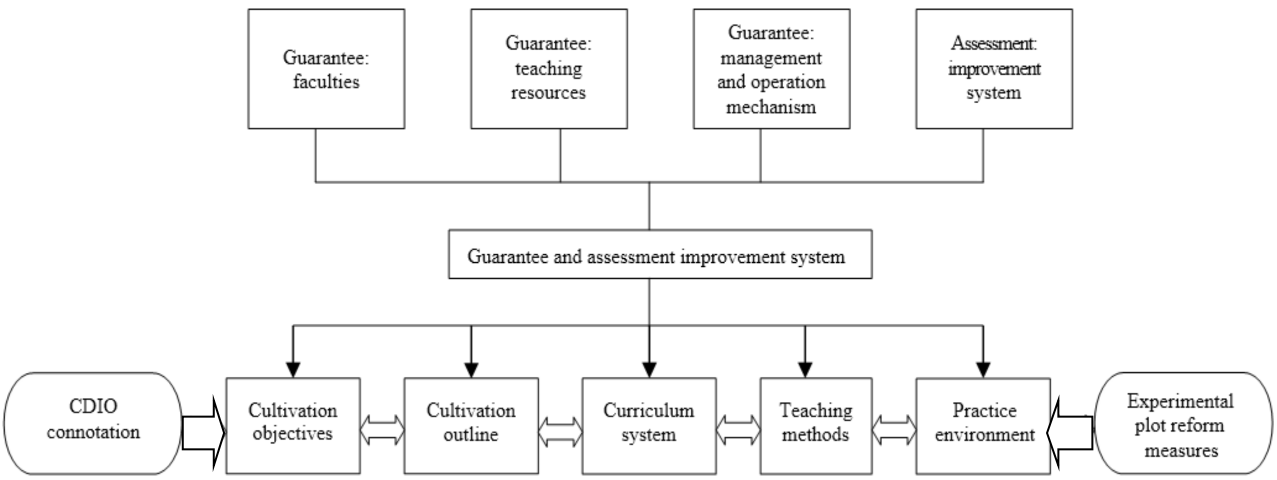


Figure 1: Software engineering training based on CDIO.

## CURRICULUM SYSTEM

It is necessary that engineering education instils disciplinary knowledge in students and that it cultivates their interpersonal skills, as well as knowledge of the products, processes and systems in an extensive field. Based on CDIO, the software engineering major was reoriented to establish a three-level integrated curriculum system. Level I is a single curriculum or curriculum group (point); Level II is professional practice and training (a line); and Level III is graduation design and comprehensive reform (a surface). This is explained further:

- Level I

  The level I single curriculum or curriculum group refers to professional basic courses and other professional courses. Students are required to grasp the basic theories and knowledge and be capable of using this knowledge. The software engineering training is based on CDIO using a micro-project.

- Level II

  Level II is concerned with project practice and practical training. The small quantity of classroom teaching is specific to project practice and practical training. Most of the teaching time is used by teachers instructing students on practical training and development practices for projects. Hundreds of well-designed small practice cases are available for student teams. A student team comprises three or four students and they complete the whole process of a project based on CDIO, including collaboration and division of labour.

- Level III

  Level III is aimed at graduation design and comprehensive reform. For the purpose of satisfying enterprise demands for software talent, graduation design projects are actual applications from software enterprises. Students make comprehensive use of the basic principles, professional knowledge and skills acquired during the four years of college study, and implement the CDIO mantra of conceive, design, implement and operate for the graduation design project. Due to these projects being actual corporate projects, the realistic process of software engineering training based on CDIO and industry-academy co-operation is fully achieved. This level of software engineering personnel training based on a CDIO curriculum is a comprehensive reform for the university and students, greatly extending the application scope of software engineering personnel training.

### Implementation of CDIO

#### Conceive

In the conceive (C) stage, students are required to deepen their understanding of basic concepts in software engineering, software development and software processes and, hence, *conceive* of a project to be researched and developed. The teaching method involves teachers providing student teams with several software cases. Each student team chooses a case to analyse, deliberate on and discuss. They will consider the software life cycle, development model and software processes applicable to the case. Hence, they acquire basic perceptual knowledge of software development, and their understanding of fundamental analysis and design methods of software engineering is deepened. After that, they can perform a feasibility study of the project.

#### Design

In the design (D) stage, the student team is required to complete the design of the software through both a division of labour and co-operation. Outputs include the design of a technical route and the project design. By way of design practice, students are able to grasp the basic principles, methods, techniques and tools of software engineering for practical applications.

#### Implement

In the implement (I) stage, the student team implements the design through division of labour and co-operation. The work is checked and tested on a regular basis, according to the project implementation plan. Great store is placed on students' pragmatism, engineering ability and team spirit.

#### Operate

In the operate (O) stage, the student team is required to make a practical application out of the developed system, and to establish corresponding application, marketing and service channels. For projects where it is difficult to make an actual application, a simulated application environment can be used to establish implementation of the simulated application.

After a project is completed, teachers and relevant interested parties assess, and summarise the project. Tracking and maintenance are part of practical applications. Operational practice is the most difficult, and its main purpose is to enable students to know the market; thereby, cultivating their pragmatism and practical working ability.

TEACHING METHODS

- Sub-project:

The sub-project is an implementation of the teaching reforms of case selection, project implementation and teamwork. The reforms conform to the CDIO outline, with a sub-project as the vehicle for learning. The software engineering major has 20 sub-projects for the three-level curriculum system suitable for CDIO teaching.

The sub-project is at the core of the engineering education reform and the aim is to stimulate the enthusiasm of teachers and students, while encouraging teachers to participate independently in various seminars about CDIO. The educational pattern is set by a teacher's sub-project, and teaching research theses are written about the outcomes; thus, promoting the development of this form of education.

- Learning by doing:

The pattern of software engineering training based on CDIO involves the establishment of pilot projects. Huge efforts are made to interconnect the education with industrial practice, by deep, broad and intensive *industry-academy* co-operation. Hence, an effective mechanism is required for the college-enterprise interaction. Communication with software enterprises establishes their actual demands for software talent, including required technology and quality. The school co-operates with enterprises, to improve students' engineering practice abilities by means, such as the establishment of a training base and joint school-enterprise management of projects. The intent is to cultivate excellent engineers having innovative and management capabilities relevant to enterprises.

The funds, talent, technology and resources required to support the cultivation of talent may be obtained from many international and Chinese channels. A modern practice base satisfying the demands for software engineering training based on CDIO should be built, and an integrated industry-university-research institute established. Curriculum teaching reform enhances the quality of the teaching. The concept that …*an enterprise is the second classroom of a school, and the university is a manpower resource, library, and preprocessing workshop for an enterprise* drives the co-operation. Dynamic adjustments should be made to curricula, according to market demands and technological developments. Hence, software engineering talent is fostered to meet enterprise demands.

- Research and teaching:

Software engineering training based on CDIO encourages teachers to improve their teaching methods, often informed by research projects that they participated in. A system of research projects can encourage the cultivation of students' innovative abilities. To optimise teaching content and reform teaching methods, teachers are encouraged to infuse research content into the teaching. Teachers are encouraged to update teaching content with new scientific developments; thereby, expanding the concepts covered by traditional teachers. A working mechanism to introduce scientific achievements into teaching encourages teachers to infuse the latest and most relevant to the industry achievements into teaching materials. Optional courses are offered reflecting the academic frontier of knowledge. Projects may include scientific research tasks, making students know and master the latest frontier knowledge. Scientific research teams are encouraged, with interactive and participatory classroom teaching, and students' independent inquiry-based study.

- Chinese national curricula and reform of software engineering based on CDIO:

Software engineering is a Chinese national level excellent curriculum in the software engineering major. Teaching reform based on CDIO began in 2007, when Neijiang Teachers College initially explored CDIO engineering education.

Carrying out software engineering personnel training based on CDIO, covering every aspect of the software engineering curriculum, inspires students' enthusiasm, widens their innovative thinking and exercises their professional competence. It fosters students' co-operative ability and perseverance. It facilitates teaching research and team construction, which play a major role in promoting the development of engineering education in the software engineering major.

THE PRACTICE ENVIRONMENT

A place to practice is a key factor and a significant resource for improving students' abilities to develop and test products, processes and systems. The university pays great attention to the construction, management and updating of the practice environment. There are fundamental laboratories; speciality fundamental laboratories, speciality laboratories and innovation laboratories. Extensive co-operation with government and others occurs; as well, a joint research and development centre and collaboration centre were built. The university makes full use of intramural resources to build an intramural practice base. Also, the university has active, extensive co-operation with well-known IT enterprises in China, to set up an extramural practice base.

According to the concept of *bringing in, going out*, the university actively introduced advanced educational concepts and the high-quality education resources of developed countries; it also established an international practice base with the co-operation of overseas companies and universities. This practice base is able to support the three-category teaching pattern of CDIO. The design of products, processes and systems, along with the implemented teaching pattern, is jointly supported by speciality laboratories, a joint research and development centre, and a collaboration and practice base. Projects of various types and diversified requirements can be implemented including:

- Course-based projects:

  Course-based projects are undertaken by a group of students majoring in a course. They spend a semester in designing and implementing the given course project in a speciality laboratory.

- Team-oriented projects:

  Team projects occur in courses dealing with professional practice and practical training. It takes one semester to one year to complete a small project.

- Collaborative projects:

  Collaborative projects are based on the actual demands of co-operative enterprises. It is required that it takes one semester to one year to complete a medium-sized project in the enterprise.

- Extracurricular projects:

  Extracurricular projects are constructed for competitions. A student team is composed of students majoring in several different specialities, and who undertake a competition programme in the joint research and development and collaboration centre. A project usually takes one semester to several years. It may be passed down from cohort to cohort, with senior students passing on the project and guiding junior students.

- Test and operation projects:

  A test and operation project enables students to experience test and operation processes, and to master the *operation* concept in an engineering system. These types of project need to be completed with the support of enterprises in the joint research and development centre.

Laboratories are used in two types of teaching. For disciplinary knowledge consolidation, the fundamental laboratory and speciality fundamental laboratory are used, and the innovation laboratory is used for knowledge discovery. The fundamental laboratory and speciality fundamental laboratory provide students with resources for actual operations and active learning, enabling them to consolidate professional knowledge.

The innovation laboratory supports students in research-oriented projects. Undergraduate research projects take place in the third and fourth year of study. They involve participation in tutors' scientific research projects or independently, choosing a research-oriented project under the guidance of a tutor. The innovation laboratory for masters and doctoral students is used mainly for supporting long-term, persistent scientific research.

## GUARANTEE AND ASSESSMENT SYSTEM

A guarantee and assessment improvement system is a quality assurance system used to monitor, control and suggest improvements in a system. Prior to developing software engineering training based on CDIO, a teaching quality guarantee system already existed. However, the traditional teaching quality guarantee system is not applicable to the new system based on CDIO.

In the first place, most of the teaching quality guarantees are macro ones from the perspective of the software engineering major. It is not reasonable for different subjects and disciplines to have the same set of quality guarantees.

In the second place, the traditional teaching quality guarantee system focuses on the monitoring of the teaching process, which is not conducive to the improvement of teachers' enthusiasm, since such a system concentrates on finding faults [7]. Therefore, it is important to establish software engineering training based on a CDIO-oriented guarantee and assessment improvement system for resolving the problems that arise.

Guarantee Systems

Guarantee systems for the teaching include:

- Guarantee system of faculties:

  Faculties in the software engineering major have part-time and full-time teachers. Full-time teachers account for 40 per cent, and part-time 60 per cent of the total. The scale, structure and level of the faculties can meet the demands for *international, application-oriented software engineering* training. Additionally, the software engineering major makes great efforts to promote teachers' dedication, professional enthusiasm and care for students. There is a spirit of competition and co-operation and of teaching others by example. Teachers are encouraged to take the initiative to impart knowledge and to cultivate students. Stress is placed on international, competitive, innovative knowledge. Talent teams are constructed to include masters to help cultivate top talent. Such talent is cultivated by implementing *faculty construction engineering of software university* in compliance with national (Chinese), provincial (ministry) and university levels.

  To make teaching standard and rigorous, the university formulates work regulations for the faculty, position appointment and assessment system. The university adheres to policies, such as students evaluating teachers and teachers (leaders) evaluating teachers. For many years, the mantra of *…rigorously doing scholarly research, strictly teaching, cherishing posts and wholehearted devotion to work* has been promoted through the teaching and research work of the whole university. A provincial CDIO teaching team was established dependent on a national Linux centre, a provincial key laboratory of software engineering, a provincial cloud computing engineering research centre and a provincial facility for top teachers.

- Guarantee of teaching resources:

  A guarantee of abundant teaching resources is an important basis for reform. To stably provide the resources required for reform on a long-term basis is basic to achieving the educational reform objectives. Therefore, efforts are made constantly to raise resources for the software engineering major, including funds and experimental facilities. Additionally, a *Scheme for Post Establishment and Employment of the Software Engineering Major* details the rigid stipulations on job responsibilities that every teacher must observe. Such a scheme provides a good basis for every teacher to satisfy their responsibilities, and stipulates mandatory tasks for every teacher; thereby, guaranteeing the operation of the team.

- Guarantee of management and operation:

  A responsibility system details daily operation and management. The teaching instruction committee of the university works as an expert committee, and is responsible for assessment, acceptance and quality guarantees. A responsibility system of project leaders details the operation and management of sub-projects and projects.

Assessment System

Teaching evaluation is used for assessing teaching processes and students' learning outcomes. An iterative software life cycle process is adopted to make improvements according to assessment results.

Traditional teaching evaluations assess every stage separately, rather than evaluating a system as an organic whole. However, software engineering training based on CDIO attaches importance to evaluating the whole teaching process. For example, if a teacher's classroom teaching is to be evaluated, attention will be paid to the obvious factors, such as teaching presentation, and more important is concern for factors, such as case teaching, project-based teaching and research-oriented teaching. It should be evaluated whether a teacher teaches projects, cases or research subjects as per the whole software life cycle and guides students to implement practice and practical training of the whole lifecycle process.

Traditional teaching evaluation is only aimed at teachers' teaching. But, the teaching evaluation must make a comprehensive assessment of teachers' teaching and students' learning outcomes. Evaluation of students' learning outcomes is based mainly on students' understanding, grasp and application of knowledge, as well as understanding of engineering practice. Only if students obviously learn and enhance their engineering capabilities through teaching, can teachers' teaching quality and teaching efficiency be viewed favourably.

The software engineering major adopted an iterative process in constructing the assessment system, i.e. software engineering personnel training based on CDIO is improved and perfected according to the evaluation results of the previous round of teaching. For example, in 2009, the software engineering major included an experimental small-scale application. After the first round of reform, the software engineering major was improved in 2011 in accordance with assessment results. The application was formally included in the software engineering training based on CDIO, and applied on a large scale for undergraduates enrolled in 2011.

After the student iteration cycle in 2015, assessment results continued to be used to perfect the educational system, and these were applicable to future students. The assessment-driven improvement coincides with the iteration cycle of four years for undergraduates; hence, guaranteeing continuity of the quality of the training.

## CONCLUSIONS

Thirty-six Chinese national pilot software schools have obtained remarkable results in innovative talent cultivation. These include: the innovative CDIO-based capability maturity model of students of Central South University; the international software engineer cultivation programme of Beijing Jiaotong University; the practical high-level software programme of Northeastern University; and the *industrial, personalised, elite* - based software engineering major of Harbin Institute of Technology. These are all software engineering programmes [8].

The Department of Computer Science of Neijiang Teachers College implemented an innovative CDIO-based model for the software engineering major. From the viewpoint of the application discipline, the main pilots using CDIO implemented by Chinese universities/colleges have focused on mechanical, electrical, chemical and civil engineering. There have been few applications to the software engineering discipline. Software engineering personnel training based on CDIO provides a reference for engineering education in the software engineering discipline.

## REFERENCES


1. Ahmad, M.O., Markkula, J. and Oivo, M., Kanban for software engineering teaching in a software factory learning environment. *World Trans. on Engng. and Technol. Educ.*, 12, **3**, 338-343 (2014).
2. Varona, D., Capretz, L.F. and Raza, A., A multicultural comparison of software engineers. *World Trans. on Engng. and Technol. Educ.*, 11, **1**, 31-35 (2013).
3. He, C., Cao, Q., Du, S. and Zhang, Y., On electronic and information engineering personnel training based on C&P-CDIO mode. *Research in Higher Educ. of Engng.*, 2, 60-63 (2013).
4. Yang, C., Huan, S. and Yang, Y., Application and management of EPI-CDIO teaching for engineering students in local undergraduate colleges. *World Trans. on Engng. and Technol. Educ.*, 13, **3**, 417-420 (2015).
5. Lin, G. and Tang, Y., Application of CDIO in practice and training of an engineering automation specialty. *World Trans. on Engng. and Technol. Educ.*, 13, **3**, 407-411 (2015).
6. Li, T., Zhang, X., Wang, X., Liu, Q., Kang, Y., Wang, W. and Zhao, N., Exploration and practice of the SE-CDIO engineering education mode. *Research in Higher Educ. of Engng.*, 1, 52-57 (2014)
7. Zhang, D., Liu, Q. and Huang, G., Research and practice on the construction of teaching quality assurance system of software engineering based on CDIO. *Computer Engng. and Science*, 33, 182-185 (2011).
8. Ji, Y. and Zong, P., College of software talents training innovation experimental zone. *Nanjing University of Posts and Telecommunications of (Social Science Edn.)*, 1, 118-120 (2011).