TEAL2.O ANALYSIS MODEL

Improving Access to Science and Technology Higher Education in Resource-Poor Institutions through an Open Platform for Technology Enabled Active Learning Environment

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
# CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>2</td>
</tr>
<tr>
<td>Scope of the Solution / What Should the System Do?</td>
<td>3</td>
</tr>
<tr>
<td>Goals / Business Logic</td>
<td>4</td>
</tr>
<tr>
<td>Non-Functional Requirements</td>
<td>5</td>
</tr>
<tr>
<td>- Product Requirements</td>
<td>5</td>
</tr>
<tr>
<td>- Organizational Requirements</td>
<td>9</td>
</tr>
<tr>
<td>- External constraints</td>
<td>13</td>
</tr>
<tr>
<td>Functional Requirements: Program Description Tool</td>
<td>14</td>
</tr>
<tr>
<td>- Method for representing and mapping competencies</td>
<td>29</td>
</tr>
<tr>
<td>- Method for representing course structures</td>
<td>35</td>
</tr>
<tr>
<td>- Method for autonomous mapping of course Learning Outcomes to Program Objectives</td>
<td>39</td>
</tr>
<tr>
<td>- Method for Classifying programs - Field/Subject area/PG/UG</td>
<td>40</td>
</tr>
<tr>
<td>- Standards and guidelines for the specification of benchmarking statements</td>
<td>44</td>
</tr>
<tr>
<td>Usage Model / Functional Requirements</td>
<td>53</td>
</tr>
<tr>
<td>User Interface Model</td>
<td>55</td>
</tr>
<tr>
<td>Translating Requirements Into Components</td>
<td>56</td>
</tr>
<tr>
<td>Data Flow</td>
<td>60</td>
</tr>
<tr>
<td>Annex I: User Stories</td>
<td>62</td>
</tr>
</tbody>
</table>
This document provides the first technical representation of the TEAL2.O solution. It focuses on clarifying the business model and logic of TEAL2.O and on stipulating the software requirements. The intended audience for this document comprises the developer teams. The document should be used in conjunction with the TEAL2.O Software Design Document.
SCOPE OF THE SOLUTION
What Should the System Do?

TEAL2.0 should be developed as a comprehensive learning environment allowing for the design and delivery of courses and other forms of training in the area of Science and Technology. The TEAL2.0 environment will have features similar to Learning Managements Systems such as Moodle or Canvas, and to online course platforms such as edX or Coursera. However, it will extend its capabilities beyond the functionality available in these systems.

In contrast, TEAL2.0 should enable customizability and collaborative development at all stages of the education design and delivery process – at the stage of developing programs, at the stage of developing courses and content, and during the learning process. This will make learning more relevant to institutional requirements and individual learners’ needs and capabilities. It will enable effective feedback collection and assessment of learning efficiency. All of these features will be wrapped in an inbuilt evaluation and quality assurance process that will rank programmes, courses, content, teachers, and students based on peer reviews and interaction with the content.

The solution should be a designed as a cloud-based, open-access collaborative teaching & learning platform. It should be open for use by any institution of higher learning. Cloud reliance does not imply the solution is going to be tailored to a specific environment public/private cloud. Instead, TEAL2.0 will fully rely on open source, avoiding any undesirable vendor lock-in phenomena.
The TEAL2.0 environment should offer:

- Access to MOOCs and other digital learning content developed around the globe
- Modularity and full customizability, allowing institutions and faculty to design programs and develop teaching content that suit local needs and constraints, including those of underrepresented learners
- Autonomous assessment and quality assurance
- Integration of open source hardware and software into the learning process to maximize learners’ exposure to technology and provide access to virtual labs and experimentation
- Enhanced opportunities for collaborative learning and teaching, allowing institutions to pool available expertise and support quality
- Learner-centered education, flexible learning pathways and co-creation on the part of learners
Product Requirements

Security, performance and usability

The main concerns when developing the TEAL2.O product requirements were security, performance and usability. Maximizing quality in this regard was of paramount concern and it dictated all technological choices.

User feedback was taken into account. User feedback and concerns mostly indicated general apprehension towards new and complex systems that would entail a steep learning curve. The situation is compounded by the Covid-19 pandemic which pushed all educational activities into online mode. The TEAL2.O platform was initially envisaged as a platform supporting Science and Technology education and thus mostly targeted at technology-savvy and enthusiastic users. In the new situation, the platform is viewed by users and target institutions as a form of replacement of classroom learning. As a result, the size and diversity of the user community has grown exponentially. This community now encompasses also users that are less comfortable with habit-breaking tech solutions. It could be expected that within 1-2 years the situation would normalize but there are strong indicators that there will be no full return to the pre-Covid status quo. Rather, online-based course design and delivery will remain a strong alternative to traditional modes of education and will be increasingly integrated into university development plans, in effect creating a truly blended course design and delivery process. For the TEAL2.O to be able to serve its new purpose, we need to address the main user concerns – namely an aversion to radically new and complex learning environments – and to avoid creating a backbone, and consequently a learning environment, that requires a long investment of time and effort on the part of both lecturers and students.

This basic non-functional requirement has prompted the team to look for ways to develop the backbone in full conformity with, and within the broader technological framework, of existing Learning Management Systems (LMS) that are widely used and already accepted by users. This would allow us to preserve the habitual user interface that users are already comfortable with, while adding new functionalities and features. The approach would also facilitate the process of involving users in co-creating and further developing the TEAL2.O network through the development of new functionalities and content. The chances of co-creation increase exponentially as familiarity with the framework increases.

At the same time, TEAL2.O is more than a traditional LMS. In addition to creating content, online classrooms and courses, it will also facilitate active collaborative teaching and learning – a feature that is novel compared to any of the existing LMSs. It will incorporate outcome-based education by evaluating progress by measuring the attainment of the outcomes, as well as rank content and processes based on actual student engagement and student attainment of the expected outcomes. There exists no platform that accommodates all of these requirements. TEAL2.O will develop such features. However, the basic elements of the backbone should be grounded in existing LMS frameworks.

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
The currently available open LMS development platforms that users have grown accustomed to and largely comfortable with are:

- Moodle
- Totara
- Canvas
- Open EdeX

Our survey of the capabilities of the above open source platforms indicated that Totara is the one that contains the highest number of required features of TEAL2.O. Of course, major features – such as mapping activities/assessments/content to LOs, collaborative learning content creation, enabling collaborative learning and customization of learning content, ranking of learning content based on attainment of LOs, and quality assurance – are missing in it. But it looked like a good starting point. Initially, therefore, we considered modelling the TEAL2.O platform on Totara. Unfortunately, however, further investigation showed that even though Totara is Moodle-based and open-source, the source code is only available to Totara partners or direct subscribers. Integration with a source that is not freely accessible to potential user institutions will derail the usability and accessibility of the TEAL2.O platform. Since Totara is itself based on Moodle, the next best available option for us was to base TEAL2.O on – and integrate it with – Moodle.

Moodle is a free and open-source learning management system written in PHP. It is the world’s most popular LMS, with over 160 million current users and over 100 thousand active sites. With an equally large variety of use-cases across those sites, the Moodle LMS is the go-to for anybody looking to run a general/ traditional e-Learning program. Moodle delivers a powerful set of learner-centric tools and collaborative learning environments that empower both teaching and learning. Because of its flexibility and scalability, Moodle has been adapted for use across education, business, non-profit, government, and many other community systems of all sizes. Moodle is freely available for all users. It can be self-hosted. Moodle will be accepted by educators and learners as it will be easy to use. Apart from being fairly familiar already, it boasts a simple and flexible user interface and well-documented resources. It is web-based and features a mobile-compatible interface, making it accessible from anywhere in the world across different web browsers or devices. It has been translated into more than 120 languages and is designed to comply with open and accessibility standards. Because Moodle is the big open-source player in the LMS space, it is supported by a massive and active community with tons of plugins and options to customize it to the required specifications. It also benefits from a lot of online documentation for help with development, support issues or questions.

All in all, the features of the TEAL1.O backbone can be developed by adopting a Moodle-based structure. This will guarantee wider user acceptance, fairly good usability, full accessibility and a great degree of security. Integration of the many TEAL2.O elements can be achieved by developing them as plugins to the fully Moodle-based and Moodle-compatible skeleton.
This technology choice will impose key non-functional requirements – namely using PHP as the Programming language for Development, and building on Moodle source code.

**Infrastructure**

- **Medium-term infrastructure support**

It is recommended to keep the TEAL2.O infrastructure that is used across development, testing and integrated deployment consistent across all participants, across all regions and across all institutions. The TEAL2.O project proposal envisages purchase of on-premises support infrastructure including the following hardware components:

<table>
<thead>
<tr>
<th>Description</th>
<th>Number of Units</th>
<th>Where</th>
<th>Specifications</th>
</tr>
</thead>
</table>
| Development Servers | 3               | VIT            | Processor: 3.0GHz, 8-Core Intel Xeon E5 with 25MB L3 cache and Turbo Boost up to 3.9GHz  
Software: Mac Server software  
Memory: 16GB (four 4GB) of 1866MHz DDR3 ECC memory  
Storage:256GB PCI e-based SSD  
Graphics: Dual AMD Fire Pro D700 graphics processors with 6GB of GDDR5 VRAM each Display  
Support: Three dual-cable 5K display, Six thunderbolt display  
Connections: USB3-4, Thunderbolt 2- 6, Dual Gigabit Ethernet, HDMI.4Ultra HD  
Input: Wired Keyboard, Wired Mouse End Point Protector (device control, file tracing, device logging, USB lockdown, DLP)  
Operating System: Mac OS Sierra |

| Storage Servers     | 3               | UoP, UoM, IITA | Eight-core Intel Xeon processor. (Xeon Silver 4110), 64GB system memory, RAID 0/1/5/6/10 with controller support up to 8GB cache,  
Hybrid storage supporting SLC, MLC & TLC flash, Storage Pool should be configured with not less than 5TB usable capacity in RAID 5 using with 1.2TB 10K SAS HDDs. (Hot spare drives are Mandatory).  
R/W ratio – 70/30 Block Size – 8K, support Async replication and related licenses should be included can replicate 32 volumes per system, HTML5 Web GUI, CLI & REST, support integration with commonly used virtualization and data archival solutions (VMware,vSphere (ESXi) vCenter; SRM, Microsoft Hyper-V) |
The indicated infrastructure should be taken into account during the development process in order to avoid hardware conflicts. This on-premises support infrastructure should be able to meet the requirements of the TEAL2.O platform for around 10 years starting from the moment when the platform is deployed.

Longer-term solutions may also include using the services of a public cloud provider. It is not expected that a switch to a public service provider would impose any constraints on the technological solution and so it will not affect the development process. The choice of long-term support infrastructure can be made in the course of exploitation, toward the end of the 10-year period.

✓ Long-term support infrastructure options

Why choose a Public Cloud Provider?

In the long run, priorities may shift and it may become more reasonable to not spend valuable time in maintaining support infrastructure as such efforts do not provide any value towards expanding the TEAL2.O network. It may also be recommended not to spend valuable time in setting up application servers, databases and keeping the software and operating system updated as again none of these activities contribute towards expanding the TEAL2.O network. If priorities are so re-aligned, it may be recommended to just focus on functional code and leave all non-functional aspects to an external provider of support infrastructure.

Based on the comparison provided below, the following long-term infrastructure options exist: Public cloud IaaS (Infrastructure as a Service), PaaS (Platform as a Service), SaaS (Software as a Service) and FaaS (Function as a Service).

Which Cloud Provider to Use?

There are several factors to be considered before deciding on a preferred Cloud Provider:

1. Completeness of the vision of Cloud Patterns
2. Ability to execute and keep up with the latest trends
3. Provision of complete coverage of services during development and beyond
4. Provision of quick starts, patterns and solutions out of the box
5. Provision of consistent management of Cloud resources regardless of whether the resources are deployed via portal, command line or SDK

6. Provision of a lightweight contained environment construct that can be used to manage separate budgets across teams, separate components and role-based access for each component

7. Provision of a consolidated hybrid environment where Cloud and on-premise components can be easily integrated, should we chose to go in that direction.

As of the current status-quo, out of all Cloud Providers, obviously AWS, Microsoft Azure and Google Cloud are the market leaders. In terms of (1) and (2) above AWS and Microsoft Azure leads well ahead of Google Cloud according to Garners Report as IaaS Provider. Both AWS and Azure provide (3) and (4) in a comparably matured way, so these requirements are not significant when choosing between AWS and Azure. For (5), (6) and (7), Microsoft Azure leads the way and AWS is still catching up. In (5) for example, ARM (Azure Resource Manager) is architected well to suit all type of deployment, whereas AWS’s Cloudformation only works if the initial deployment is done via Cloudformation; thus, for new users of AWS this creates steep learning curve. In (6), Azure Subscriptions and Azure Resource Group provide a lightweight construct to manage separate budgets and separate contained environments, while the AWS account provides the same but comes with the overhead of managing multiple accounts. In (7), Microsoft born on-premise and now focusing on moving to Cloud, the ecosystem in Azure caters for most of the on-premise scenarios and even provides the capabilities and services to host applications and databases on-premise and expose them securely into Azure environments. AWS on the other hand is born in Cloud and is hence focusing only on migration from on-premise to Cloud rather than on running Hybrid solutions.

On the basis of these considerations, the more reasonable choice would be Microsoft Azure – with an account for the entire project/platform, a subscription for each partner so that it can manage its budgets and a resource group for each component in the component test environment. However, since we are discussing a long-term solution and technological options are quick to change, a similar analysis needs to be made around the 6-7 year mark after the deployment of the TEAL2.0 platform.

☐ Organizational requirements

Tools for development

Developer teams are required to use the following development tools:

Git

Installing Git on your computer and setting up can be found here. It will be used for version control.
Apache + PHP
Based on your system, you can install Apache2 and PHP.

- Debian: [LAMP](#)
- OS X: [Mountain Lion Server](#)
- Windows: [Apache2 + PHP](#)
- PHP Extensions and libraries
- Enable SSL for Apache ([Transitioning to HTTPS](#))

More information: [Installation](#)

Database
Based on you system, setup [MySQL Database](#) for Moodle.

Moodle
Manual installation: [Quick guide](#)

- Moodle Development Kit (MDK)

Only for Linux and MacOS (not available for Windows).

- [Setup](#)
  - Sample to create some instances

```bash
mdk create --install --engine mysql --run mindev users makecourse -n stable_master-mysql
```

- Docker

Docker containers provide significant advantages in software development. Docker is an open source platform which is designed to be lightweight and simple. A Docker container is a standardized unit of software used to deploy applications. The containers allow a developer to wrap up an application with all essential parts, i.e. libraries, dependencies, and ship it all out as one package. By doing this, the developer can be assured that the application will run on any machine, regardless of any customized settings. Containers also work in isolation from each other allowing a range of tasks to occur independently.

- IDE

You can setup any of the following IDE and use it for development:

- [Sublime](#)
- [PhpStorm - Setting up PhpStorm](#) for Moodle development.
More details about Moodle development environment setup can be found [HERE](#).

There is a [YouTube channel](#) that shows clips with guides and tips for the Moodle setup.

**Support and Collaboration Tools**

In a distributed environment, the building of a significantly large platform requires a lot of collaboration and consistent feedback exchange among participants. In the environment needed to create TEAL2.O work is distributed across several regions and several institutions within each region. This setup requires active collaboration during the design and implementation processes. This will create more engagement among all parties and we will get the best out of all developers, while also applying the latest trends, best practices and technology choices that will contribute to the overall quality of the project and will increase the motivation of the participants.

The types of collaborative tools that will be employed in the TEAL2.O development process are:

**Documentation and General Task Allocation**

Cloud-based Atlassian confluence will be used as it is easy to sign up and it is using matured Wiki-style documentation with concurrent editing enabled. It is Cloud-based, so no maintenance is required.

**Software Development Task allocation**

The work packages, composed of features, large-sized stories to small stories and individual tasks, all these can be allocated to an individual in any team, so they can be tracked. The cloud-based Atlassian JIRA provides clear decomposition from a large domain model to small tasks and can still maintain the hierarchy of all items.

**Delivery Methodology**

In an uncertain situation, it is recommended to start small and iterate through rather than trying to design everything upfront. It is also absolutely important for all stakeholders to be kept informed of the development of the platform and all tracking towards their final vision.

Therefore, the developer teams will use **Agile, especially Scrum, methodology, with two-week Sprint across all teams, with Sprint start and end dates aligned**. It is also proposed to include the design and refinement work in the Sprint n - 1 and integration and testing work in Sprint n + 1 for the work that is happening in Sprint n.

---

*This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.*
Source Control

All source code needs to be maintained in a collection of repositories, so that it is accessible by all participants and everyone gets to contribute towards high-quality code and best coding practices.

It is also important that all code commits are reviewed by at least another person so there is less chance of human error and so that coding consistency is maintained throughout the platform.

We will keep at least two branches of source code. One branch will be used for the stable version and the other one will be for the daily build. The stable version will be maintained by a person after the review of the current version at the end of sprint or per period.

Therefore, the developer teams will use Git-based Project for the entire platform and repositories for each domain/component/feature.

Continuous Integration and Continuous Deployment (CICD)

All developed code needs to be continuously integrated with the application server and database and deployed in an environment that is the same as the final environment. This provides early insight into any software bugs and required changes and reduces the risk of deviation from the ultimate vision.

Therefore, the developer teams will use cloud-based CICD tooling. It does not require any management other than setting up your own pipelines.

Development Environment

A development environment needs to be dedicated to each component development team so they can continuously develop and test their code without interfering with other teams’ development activities.

Therefore, the developer teams will use cloud-based contained environment for each component development team, as well as managed roles and groups that are respectively associated with each component and team. This will give flexibility for a team to be involved in one or many components.

Component Test Environment

The component test environments are needed to make sure any component can be deployed and tested in the same way as it is developed, and to make sure that there are no environment-related issues present before it gets promoted to next level for an end-to-end testing.

Therefore, the developer teams will use cloud-based contained environment per each component across the platform, so this is always kept functional and updated during the development process.

Integrated Environment

This is the final environment where all components expected to be integrated and all the completed components of the entire platform are always kept functional. It is expected that this environment
will act as a gluing environment over the component environments unless a copy of component environments is required for different types of testing such as user acceptance, partnership testing, performance testing, etc.

Therefore, the developer teams will use **cloud-based contained environment with specific roles assigned for each type of activity and these roles may or may not be assumed by the developers who developed the component.** It is expected that most often a nominated group of people will manage this (instead of the original developers) so it is considered from the user point of view.

### External constraints

The main external constraint for the development teams is the need to comply with Moodle standards and documentation. This is absolutely necessary as the idea is to develop all features as Moodle plugins. This will improve their usability and user acceptance. Developer teams will be required to comply with these constraints (consult section “Moodle Integration” in the TEAL2.0 Software Design Model).
Method for representing standards for the specification of Program Objectives (POs) in Alignment with Accreditation accords - Washington Accord/Sydney Accord/Seoul Accord

About program specifications, program objectives and learning outcomes

A program specification contains the aims, objectives or learning outcomes, program content, learning and teaching methods, process and criteria for assessment, usually with indicative reading or other reference material as well as identifying the modules or subunits of the program, setting out core and optional elements, precursors and levels.

Program specifications should make explicit the intended outcomes in terms of knowledge, understanding, skills and other attributes. They should help students to understand the teaching and learning methods that enable the outcomes to be achieved; the assessment methods that enable achievement to be demonstrated; and the relationship of the program and its study elements to the qualifications framework and to any subsequent professional qualification or career path.

Program specifications must give detail information about the program of study and typically include:

- Educational aims of the program
- Intended learning outcomes
- Program structure and requirements
- Support for students and their learning
- Criteria for admission
- Methods for evaluation and enhancing the quality and standards of teaching and learning
- Regulation of assessment
- Indicators of quality and standards
- List of mandatory and option modules
- Capabilities (skills) map

Institutions of higher education (HE) should develop, for each program they offer, a ‘program specification’ which identifies potential stopping-off points and gives the intended outcomes of the program in terms of:

- knowledge and understanding that a student will be expected to have upon completion
- key skills: communication, numeracy, use of information technology and learning how to learn;
- cognitive skills, such as an understanding of methodologies or ability in critical analysis;
- subject specific skills – such as laboratory skills.
**Objective**

A very general statement about the larger goals of the course or program. Program learning goals and objectives describe what students will learn, what skills they develop, what experiences they will have (or likely have) as a result of completing the requirements for the degree program.

**Outcome**

A very specific statement that describes exactly what a student will be able to do in some measurable way. A competency may have several specific learning outcomes so a course typically contains more outcomes than competencies.

**Competency**

A general statement detailing the desired knowledge and skills of student graduating from our course or program.

Objectives are intended results of learning process activities. Outcomes are achieved and measurable results of what was learned.

The program specifications should include clearly the learning outcomes [4].

**Standards for the specification of Program Objectives (POs) in different accreditation accords**

The standards and guidelines for engineering programs in higher education are generally given by the International Engineering Alliance (IEA). IEA is an umbrella organization for six multilateral agreements which establish and enforce amongst their members internationally benchmarked standards for engineering education and what is termed “entry level” competence to practice engineering. The Alliance, which currently has lead engineering organizations from 23 nations as
members (including five G8 and 11 G20 nations), is expanding steadily with China being the latest to apply.

The IEA covers the Washington, Sydney and Dublin Accords.

These accords recognize that the members of the European Network for Accreditation of Engineering Education (ENAAE) operate similar accreditation processes to similar standards within Europe, and its authorized members provide the Eur-ACE label to accredited programs. There is a formal mechanism between the IEA and ENAAE to maximize mutual understanding and potential benefits of the two organizations.

- [https://www.ieagreements.org/about-us/key-documents/](https://www.ieagreements.org/about-us/key-documents/)

The IEA’s core activities are:

- Consistent improvement of standards and mobility
- Defining standards of education and professional competence
- Assessment of education accreditation and evaluation of competence
- Participation in activities that are driven from the engineering profession.

*The Washington Accord*


The Washington Accord is an international accreditation agreement for undergraduate professional engineering academic degrees between the bodies responsible for accreditation in its signatory countries and regions. It was established in 1989. Its full signatories are Australia, Canada, China, CostaRica, HongKong, India, Ireland, Japan, Korea, Malaysia, New Zealand, Pakistan, Peru, Philippines, Russia, Singapore, South Africa, Sri Lanka, Taiwan, Turkey, the United Kingdom and the United States.

Usually, the Washington accord deals with the professional engineer qualification.

The Washington Accord Knowledge Profile has eight elements

<p>| WK1 | A systematic, theory-based understanding of the natural sciences applicable to the discipline. |
| WK2 | Conceptually-based mathematics, numerical analysis, statistics and formal aspects of computer and information science to support analysis and modelling applicable to the discipline. |
| WK3 | A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline. |
| WK4 | Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline. |</p>
<table>
<thead>
<tr>
<th>WK5</th>
<th>Knowledge that supports engineering design in a practice area.</th>
</tr>
</thead>
<tbody>
<tr>
<td>WK6</td>
<td>Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.</td>
</tr>
<tr>
<td>WK7</td>
<td>Comprehension of the role of engineering in society and identified issues in engineering practice in the discipline: ethics and the professional responsibility of an engineer to public safety; and the impacts of engineering activity – economic, social, cultural, environmental and sustainability.</td>
</tr>
<tr>
<td>WK8</td>
<td>Engagement with selected knowledge in the research literature of the discipline.</td>
</tr>
</tbody>
</table>

The Washington Accord Graduate Attribute Profile has 12 elements, supported by the knowledge profile:

<p>| Engineering knowledge | WA1: Apply knowledge of mathematics, natural science, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to the solution of complex engineering problems. |
| Problem analysis       | WA2: Identify, formulate, research literature and analyses complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences (WK1 to WK4). |
| Design/development of solutions | WA3: Design solutions for complex engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health, and safety, cultural, societal and environmental considerations (WK5). |
| Investigation          | WA4: Conduct investigations of complex problems using research-based knowledge (WK8) and research methods including design of experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions. |
| Modern tool usage      | WA5: Create, select and apply appropriate techniques, resources and modern engineering and IT tools, including prediction and modelling, to complex engineering problems, with an understanding of the limitations (WK6). |
| The engineer and society | WA6: Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice and solutions to complex engineering problems (WK7). |
| Environment and sustainability | WA7: Understand and evaluate the sustainability and impact of professional engineering work in the solution of complex engineering problems in societal and environmental contexts (WK7). |
| Ethics                | WA8: Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice (WK7). |</p>
<table>
<thead>
<tr>
<th>Individual and teamwork</th>
<th>WA9: Function effectively as an individual, and as a member or leader in diverse teams and in multi-disciplinary settings.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
<td>WA10: Communicate effectively on complex engineering activities with the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.</td>
</tr>
<tr>
<td>Project management and finance</td>
<td>WA11: Demonstrate knowledge and understanding of engineering management principles and economic decision-making and apply these to one’s own work as a member and leader in a team, to manage projects and in multidisciplinary environments.</td>
</tr>
<tr>
<td>Life-long learning</td>
<td>WA12: Recognize the need for, and have the preparation and ability to engage in, independent and life-long learning in the broadest context of technological change.</td>
</tr>
</tbody>
</table>

**The Sydney Accord**


The Sydney Accord covers engineering technologist qualifications.

The Sydney Accord was signed in June 2001, by seven founding signatories representing: Australia, Canada, Hong Kong, Ireland, New Zealand, United Kingdom and South Africa.

As with the other Accords, the signatories are committed to development and recognition of good practice in engineering education. The Sydney Accord is specifically focused on academic programs dealing with engineering technology.

Currently there are eleven signatories that make up the Sydney Accord. There are also two organisations that hold provisional signatory status.

**Signatories of the Sydney Accord**

- Australia - (Engineers Australia, 2001)
- Canada - ([Canadian Council of Technicians and Technologists](https://www.cctt.ca), 2001)
- Taiwan - ([Institute of Engineering Education Taiwan](https://www.ieet.org.tw), 2014)
- Hong Kong - ([The Hong Kong Institution of Engineers](https://www.hkie.org.hk), 2001)
- Ireland - ([Engineers Ireland](https://www.engineersireland.ie), 2001)
- Korea - (Accreditation Board for Engineering Education of Korea, 2013)
- Malaysia - ([Board of Engineers Malaysia](https://www.bem.gov.my), 2018)
- New Zealand - ([Institution of Professional Engineers New Zealand](https://www.ipenzn.org.nz), 2001)
- South Africa - ([Engineering Council of South Africa](https://www.ecsa.org.za), 2001)
- United Kingdom - ([Engineering Council UK](https://www.engc.org.uk), 2001)
- United States - ([ABET](https://www.abet.org), 2009)
- Sri Lanka - ([The Institution of Engineers Sri Lanka](https://www.iesl.org.lk), 2001)
The Seoul Accord


The Seoul Accord is a multi-lateral agreement among agencies responsible for accreditation or recognition of tertiary-level computing and IT-related qualifications.

Signatories: Australia, Canada, Hong Kong, Korea, Japan, United Kingdom, Taiwan, USA, Sri Lanka, New Zealand, Ireland, Malaysia, Mexico.

The Seoul Accord graduate attributes for Computing Professional, comparative to other two qualifications, are:

<table>
<thead>
<tr>
<th>Differentiating Characteristic</th>
<th>... for Seoul Accord (Computing Professional) Graduate</th>
<th>... for Computing Technologist Graduate</th>
<th>... for Computing Technician Graduate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Academic Education</td>
<td>Completion of an accredited program of study designed to prepare graduates as computing professionals</td>
<td>Completion of a program of study typically of shorter duration than for professional preparation</td>
<td>Completion of a program of study typically of shorter duration than for technologist preparation</td>
</tr>
<tr>
<td>Knowledge for Solving Computing Problems</td>
<td>Breadth and depth of education and type of knowledge, both theoretical and practical</td>
<td>Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements</td>
<td>Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to defined and applied computing procedures, processes, systems, or methodologies</td>
</tr>
<tr>
<td>Problem Analysis</td>
<td>Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions</td>
<td>Identify, formulate, research literature, and solve broadly-defined computing problems reaching</td>
<td>Identify and solve well-defined computing problems reaching substantiated conclusions using</td>
</tr>
<tr>
<td>Design/Development of Solutions</td>
<td>Breadth and uniqueness of computing problems, i.e., the extent to which problems are original and to which solutions have previously been identified or codified</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design solutions for broadly-defined computing technology problems, and contribute to the design of systems, components, or processes to meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design solutions for well-defined computing problems, and assist with the design of systems, components, or processes to meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Modern Tool Usage</th>
<th>Level and appropriateness of the tool to the type of activities performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations</td>
<td></td>
</tr>
<tr>
<td>Select and apply appropriate techniques, resources, and modern computing tools to broadly-defined computing activities, with an understanding of the limitations</td>
<td></td>
</tr>
<tr>
<td>Apply appropriate techniques, resources, and modern computing tools to well-defined computing activities, with an awareness of the limitations</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Individual and Team Work</th>
<th>Role in, and diversity of, the team</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings</td>
<td></td>
</tr>
<tr>
<td>Function effectively as an individual and as a member or leader in diverse technical teams</td>
<td></td>
</tr>
<tr>
<td>Function effectively as an individual and as a member in diverse technical teams</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Communication</th>
<th>Level of communication according to type of activities performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communicate effectively with the computing community and with society at large about complex computing activities by being able to</td>
<td></td>
</tr>
<tr>
<td>Communicate effectively with the computing community and with society at large about broadly-defined computing activities by being able to</td>
<td></td>
</tr>
<tr>
<td>Communicate effectively with the computing community and with society at large about well-defined computing activities by being able to</td>
<td></td>
</tr>
</tbody>
</table>

---

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
<table>
<thead>
<tr>
<th>Computing Professionalism and Society</th>
<th>No differentiation in this characteristic except level of practice</th>
<th>Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice</th>
<th>Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to computing technologist practice</th>
<th>Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to computing technician practice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethics</td>
<td>No differentiation in this characteristic except level of practice</td>
<td>Understand and commit to professional ethics, responsibilities, and norms of professional computing practice</td>
<td>Understand and commit to professional ethics, responsibilities, and norms of computing technologist practice</td>
<td>Understand and commit to professional ethics, responsibilities, and norms of computing technician practice</td>
</tr>
<tr>
<td>Life-long Learning</td>
<td>No differentiation in this characteristic except level of practice</td>
<td>Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional</td>
<td>Recognize the need, and have the ability, to engage in independent learning for continual development as a computing technologist</td>
<td>Recognize the need, and have the ability, to engage in independent learning for continual development as a computing technician</td>
</tr>
</tbody>
</table>

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
The ENAEE – European Network for Accreditation of Engineering Education

- [https://www.enaee.eu/eur-ace-system/](https://www.enaee.eu/eur-ace-system/)

Under ENAEE, in 2014 a Mutual Agreement was signed for the recognition of engineering degree programs.

The framework “comprises three cycles (including, within national contexts, the possibility of intermediate qualifications), generic descriptors for each cycle based on learning outcomes, and credit ranges in the first and second cycles”.

The overall result of the application of the EQF is a range of Bachelor and Master Degree programs in engineering now offered in European Higher Education Institutions. These are described here in terms of the European Credit Transfer System as follows:

a) Fulltime Bachelor degree programs in engineering are now of 180, 210 or 240 ECTS credits.

b) Fulltime Master degree programs in engineering are of 60, 90 or 120 ECTS credits.

As established by the “Recommendation of the European Parliament and of the Council” of 23 April 2008, the descriptor for the first cycle in the Framework for Qualifications of the European Higher Education Area (Bologna process) corresponds to the learning outcomes for the EQF, level 6. The descriptor for the second cycle in the Framework for Qualifications of the European Higher Education Area corresponds to the learning outcomes for the EQF, level 7.

The Program Outcomes are described here separately for both Bachelor and Master Degree programs with reference to the following eight learning areas:

- Knowledge and understanding;
- Engineering Analysis;
- Engineering Design;
- Investigations;
- Engineering Practice;
- Making Judgements;
- Communication and Team-working;
- Lifelong Learning.

Conclusions

Graduate attributes and professional competencies are mutually agreed by all accords – Washington, Sydney and Dublin, under the umbrella of IEA. The European Network for Accreditation of Engineering Education (ENAEE) operates with similar standards within Europe.

- [http://www.ieagreements.org](http://www.ieagreements.org)
The Seoul accord deals with the tertiary-level computing and IT-related qualifications, so it is independent from the above accords.

Typical engineering activity requires several roles including those of the (professional) engineer, engineering technologist and engineering technician, recognized as professional registration categories in many jurisdictions. Each qualification has common standards with the others and different requirements for the competencies.

<table>
<thead>
<tr>
<th>Differentiating Characteristic</th>
<th>Professional Engineer</th>
<th>Engineering Technologist</th>
<th>Engineering Technician</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comprehend and apply universal knowledge: Breadth and depth of education and type of knowledge</td>
<td>EC1: Comprehend and apply advanced knowledge of the widely-applied principles underpinning good practice</td>
<td>TC1: Comprehend and apply the knowledge embodied in widely accepted and applied procedures, processes, systems or methodologies</td>
<td>NC1: Comprehend and apply knowledge embodied in standardized practices</td>
</tr>
<tr>
<td>Comprehend and apply local knowledge: Type of local knowledge</td>
<td>EC2: Comprehend and apply advanced knowledge of the widely-applied principles underpinning good practice specific to the jurisdiction in which he/she practices.</td>
<td>TC2: Comprehend and apply the knowledge embodied procedures, processes, systems or methodologies that is specific to the jurisdiction in which he/she practices.</td>
<td>NC2: Comprehend and apply knowledge embodied in standardized practices specific to the jurisdiction in which he/she practices.</td>
</tr>
<tr>
<td>Problem analysis: Complexity of analysis</td>
<td>EC3: Define, investigate and analyze complex problems</td>
<td>TC3: Identify, clarify, and analyze broadly-defined problems</td>
<td>NC3: Identify, state and analyze well-defined problems</td>
</tr>
<tr>
<td>Design and development of solutions: Nature of the problem and uniqueness of the solution</td>
<td>EC4: Design or develop solutions to complex problems</td>
<td>TC4: Design or develop solutions to broadly-defined problems</td>
<td>NC4: Design or develop solutions to well-defined problems</td>
</tr>
<tr>
<td>Evaluation: Type of activity</td>
<td>EC5: Evaluate the outcomes and impacts of complex activities</td>
<td>TC4: Evaluate the outcomes and impacts of broadly defined activities</td>
<td>NC5: Evaluate the outcomes and impacts of well-defined activities</td>
</tr>
<tr>
<td>Protection of society: Types of activity and responsibility to public</td>
<td>EC6: recognize the reasonably foreseeable social, cultural and environmental effects of complex activities generally, and have regard to the need for</td>
<td>TC6: Recognize the reasonably foreseeable social, cultural and environmental effects of broadly-defined activities generally, and have regard to the need for</td>
<td>NC6: Recognize the reasonably foreseeable social, cultural and environmental effects of well-defined activities generally, and have regard to the need for</td>
</tr>
</tbody>
</table>

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
<table>
<thead>
<tr>
<th>Requirement</th>
<th>EC7: Meet all legal and regulatory requirements and protect public health and safety in the course of his or her activities</th>
<th>TC7: Meet all legal and regulatory requirements and protect public health and safety in the course of his or her activities</th>
<th>NC7: Meet all legal and regulatory requirements and protect public health and safety in the course of his or her activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Legal and regulatory:</td>
<td>EC8: Conduct his or her activities ethically</td>
<td>TC8: Conduct his or her activities ethically</td>
<td>NC8: Conduct his or her activities ethically</td>
</tr>
<tr>
<td>No differentiation in this characteristic</td>
<td>EC9: Manage part or all of one or more complex activities</td>
<td>TC9: Manage part or all of one or more broadly-defined activities</td>
<td>NC9: Manage part or all of one or more well-defined activities</td>
</tr>
<tr>
<td>Ethics:</td>
<td>EC10: Communicate clearly with others in the course of his or her activities</td>
<td>TC10: Communicate clearly with others in the course of his or her activities</td>
<td>NC10: Communicate clearly with others in the course of his or her activities</td>
</tr>
<tr>
<td>No differentiation in this characteristic</td>
<td>EC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>TC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>NC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
</tr>
<tr>
<td>Manage engineering activities: Types of activity</td>
<td>EC12: Be responsible for making decisions on part or all of complex activities.</td>
<td>TC12: Be responsible for making decisions on part or all of one or more broadly defined activities.</td>
<td>NC12: Be responsible for making decisions on part or all of one or more well-defined activities.</td>
</tr>
<tr>
<td>Lifelong learning:</td>
<td>EC13: Recognize complexity and assess alternatives in light of competing requirements and incomplete knowledge. Exercise sound judgement in the course of his or her complex activities.</td>
<td>TC13: Choose appropriate technologies to deal with broadly defined problems. Exercise sound judgement in the course of his or her broadly-defined activities.</td>
<td>NC13: Choose and apply appropriate technical expertise. Exercise sound judgement in the course of his or her well-defined activities.</td>
</tr>
<tr>
<td>Preparation for and depth of continuing learning.</td>
<td>TC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>TC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>NC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
</tr>
<tr>
<td>Judgement: Level of developed knowledge, and ability and judgement in relation to type of activity</td>
<td>TC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>TC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>NC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
</tr>
<tr>
<td>Responsibility for decisions: Type of activity for which responsibility is taken</td>
<td>TC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>TC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
<td>NC11: Undertake CPD activities sufficient to maintain and extend his or her competence</td>
</tr>
<tr>
<td>The Washington Accord program</td>
<td>The Sydney Accord program</td>
<td>The ENAEE European guidelines</td>
<td></td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------------------------</td>
<td>-----------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>Engineering knowledge</strong></td>
<td>WA1: Apply knowledge of mathematics, natural science, engineering fundamentals and an engineering specialisation as specified in WK1 to WK4 respectively to the solution of complex engineering problems.</td>
<td>SA1: Apply knowledge of mathematics, natural science, engineering fundamentals and an engineering specialisation as specified in SK1 to SK4 respectively to defined and applied engineering procedures, processes, systems or methodologies.</td>
<td>EN1: knowledge and understanding of the mathematics and other basic sciences underlying their engineering specialisation</td>
</tr>
<tr>
<td><strong>Problem analysis</strong></td>
<td>WA2: Identify, formulate, research literature and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.</td>
<td>SA2: Identify, formulate, research literature and analyse broadly-defined engineering problems reaching substantiated conclusions using analytical tools appropriate to the discipline or area of specialisation.</td>
<td>EN2: identify, formulate and solve engineering problems in their field of study; analyse complex engineering products, processes and systems in their field of study; to select and apply relevant methods from established analytical, computational and experimental methods; correctly interpret the outcomes of such analyses</td>
</tr>
<tr>
<td><strong>Design / development of solutions</strong></td>
<td>WA3: Design solutions for complex engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.</td>
<td>SA3: Design solutions for broadly-defined engineering technology problems and contribute to the design of systems, components or processes to meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.</td>
<td>EN3: develop and design complex products (devices, artefacts, etc.), processes and systems in their field of study to meet established requirements, that can include an awareness of non-technical – societal, health and safety, environmental, economic and industrial – considerations; to select and apply relevant design methodologies</td>
</tr>
<tr>
<td><strong>Investigations</strong></td>
<td>WA4: Conduct investigations of complex problems using research-based knowledge and</td>
<td>SA4: Conduct investigations of broadly-defined problems; locate, search and select relevant data from codes,</td>
<td>EN4: consult and critically use scientific databases and other appropriate sources of information, carry out</td>
</tr>
<tr>
<td>Modern tool usage / practice</td>
<td>WAS: Create, select and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, to complex engineering problems, with an understanding of the limitations.</td>
<td>SA5: Select and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, to broadly-defined engineering problems, with an understanding of the limitations. (SK6)</td>
<td>EN5: understanding of applicable techniques and methods of analysis, design and investigation and of their limitations in their field of study; • laboratory/workshop skills to design and conduct experimental investigations, interpret data and draw conclusions in their field of study.</td>
</tr>
<tr>
<td>Communication, individual and team work</td>
<td>WA6: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions. Function effectively as an individual, and as a member or leader in diverse teams and in multi-disciplinary</td>
<td>SA6: Communicate effectively on broadly-defined engineering activities with the engineering community and with society at large, by being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions Function effectively as an individual, and as a member or leader in diverse teams.</td>
<td>EN6: communicate effectively information, ideas, problems and solutions with engineering community and society at large; • function effectively in a national and international context, as an individual and as a member of a team and to cooperate effectively with engineers and non-engineers.</td>
</tr>
</tbody>
</table>
Lifelong learning

**WA7**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**SA7**: Recognize the need for, and have the ability to engage in independent and life-long learning in specialist technologies.

**EN7**: recognise the need for and to engage in independent life-long learning; follow developments in science and technology.

The program objectives (POs) should be classified in three categories and further subdivided in the 7/8 learning areas, as presented in the following scheme.

<table>
<thead>
<tr>
<th>Programme objectives</th>
<th>Learning Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge</td>
<td></td>
</tr>
<tr>
<td>1. Engineering</td>
<td>2. Problem analysis</td>
</tr>
<tr>
<td>Knowledge</td>
<td>3. Design and development of solutions</td>
</tr>
<tr>
<td>5. Modern tool usage / practice</td>
<td>4. Investigation</td>
</tr>
<tr>
<td>Skills</td>
<td></td>
</tr>
<tr>
<td>6. Communication and teamwork</td>
<td></td>
</tr>
<tr>
<td>Competencies</td>
<td></td>
</tr>
<tr>
<td>7. Lifelong learning</td>
<td></td>
</tr>
</tbody>
</table>

**PO/LO**

Learning outcomes (LOs) have been defined as a statement of what a learner is expected to know, understand, or be able to do at the end of a learning process.

At a basic level the learning outcomes may require learners to be able to define, recall, list, describe, apply, explain or discuss. For a more advanced program the learners may be expected to be able to formulate, analyse, appraise, evaluate, estimate, develop or construct. The verb will usually be followed by words indicating on what or with what the learner is acting and the nature or context of
the performance required as evidence that the learning was achieved. These additional words also indicate the level of learning achieved.

The EQF classifies learning outcomes into knowledge (facts, principles and concepts), skills (cognitive and practical) and competences (such as ability to take responsibility and show autonomy). In some settings the EQF categories are subdivided further. For example, competence is divided into personal and social competences and in another case: context, role, learning to learn and insight.

<table>
<thead>
<tr>
<th>Specific</th>
<th>• Are clear and specific</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measurable</td>
<td>• Can be measured through assessment/evaluation</td>
</tr>
<tr>
<td>Achievable</td>
<td>• The learning is achievable in one semester/time allocated for the course</td>
</tr>
<tr>
<td>Relevant</td>
<td>• Relevant in relation to the program/course description and field/discipline</td>
</tr>
<tr>
<td>Time-limited</td>
<td>• Can be achieved within the time available in the semester</td>
</tr>
</tbody>
</table>

LOs must be SMART [5]

References

Method for representing and mapping competencies

**Taxonomy based**

Blooms/SOLO

Bloom’s Taxonomy is a set of taxonomies in three domains of learning, namely the cognitive, affective and psychomotor [1, 2]. The Cognitive Domain in Bloom’s Taxonomy consists of 6 cognitive levels of complexity: Knowledge, Comprehension, Application, Analysis, Synthesis and Evaluation. The first three levels are generally referred to as lower levels of thinking and the last three are referred to as higher levels of thinking [1].

The learning processes corresponding to each level can be briefly summarized as:

1. Knowledge - an ability to recall and remember information;
2. Comprehension – an ability to understand and explain concepts;
3. Application – an ability to use information in a new setting;
4. Analysis – an ability to analyse and distinguish parts;
5. Synthesis – an ability put things together and develop a new product; and
6. Evaluation – an ability to judge and justify a decision or point of view.

Bloom’s Taxonomy revised in 2000 and the metacognitive knowledge dimension was added. The nouns changed to verbs with the last two cognitive processes switched in the order.

- Remember
- Understand
- Apply
- Analyse
- Evaluate
- Create

Due to its relation to the cognitive abilities, Bloom’s Taxonomy can be applied in any discipline. Multiple studies have attempted to integrate and apply Bloom’s Taxonomy to the curriculum design to enhance engineering courses or evaluate students’ learning [1, 2]. In many cases Bloom’s Taxonomy is applied to the development of course learning outcomes. The CDIO Initiative, for example, which is implemented by more than 100 engineering schools worldwide, implements Bloom’s Taxonomy for the curriculum and learning outcome development [3].

Bloom’s Taxonomy provides a common language to compare and discuss two different subject areas. It helps understand how these areas overlap or can deliver conceptual and practical knowledge concurrently [3]. It also helps to assess the cognitive aspects of the design activities. This makes the taxonomy a good tool for the development of a transdisciplinary design curriculum with emphasis on cognitive development.

Since engineering design education is mostly based on teaching students the basics of engineering and then combining this knowledge in design projects at later stages, a successful synthesis of knowledge
from first-year to last year courses is essential and can be achieved through application of Bloom’s Taxonomy as a common language or unifying foundation. In our case we aim to develop a first-year engineering design course that incorporates Bloom’s taxonomy and knowledge from other engineering disciplines in order to be applicable to all departments’ curriculums and facilitate students’ success in engineering design throughout the later years of their undergraduate studies.

**SOLO**

A much less known taxonomy of assessing student learning is SOLO, which was created by John Biggs and Kevin Collis in 1982. According to Biggs, “SOLO, which stands for the Structure of the Observed Learning Outcome, is a means of classifying learning outcomes in terms of their complexity, enabling us to assess students’ work in terms of its quality not of how many bits of this and of that they got right.”

According to Biggs and Collis (1982), there are five stages of “ascending structural complexity.” Those five stages are:

- **Prestructural** - incompetence (they miss the point).
- **Unistructural** - one relevant aspect
- **Multistructural** - several relevant and independent aspects
- **Relational** - integrated into a structure
- **Extended Abstract** - generalized to new domain

One of the positives sides to SOLO is that it makes it easier for teachers to identify the levels, and therefore help guide students through the learning process.

John Hattie is a proponent of SOLO Taxonomy, and has broken it down to an easier way for students to understand which gives them the ability to assess their own learning. Hattie suggests that teachers can use:

- **No Idea** - equivalent to the prestructural level.
- **One Idea** - equivalent to the unistructural level
- **Many Ideas** - equivalent to the multistructural level
- **Relate** - equivalent to the relational level
- **Extend** - equivalent to the extended abstract

These advantages concern not only item construction and scoring, but incorporate features of the process of evaluation that pay attention to how students learn, and how teachers devise instructional procedures to help students use progressively more complex cognitive processes. Both teachers and students often progress from surface to deeper constructs and this is mirrored in the levels of the SOLO taxonomy.
VALIDATION OF PLANNED REQUIREMENTS

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Solo taxonomy [2, 3]

Bloom’s taxonomy [3]
There are several verbs which can be used to define learning outcomes for the different levels of Bloom’s and Solo taxonomy (see the figures above). These refer to the behaviour/performance.

Unlike the experience of some with the Bloom taxonomy, it is relatively easy to identify and categorize the SOLO levels.

Similarly, teachers could be encouraged to use the ‘plus one’ principle when choosing appropriate learning material for students. That is, the teacher can aim to move the student one level higher in the taxonomy by appropriate choice of learning material and instructional sequencing.

Based on technical competencies

- Specification of the breadth and depth of the expected competency
- Mapping depth and breadth of competencies to Program Objectives

In the field of education, concept maps are used as both a learning tool and an evaluation tool to assess student learning. The concept map can be used to:

- Foster meaningful learning by helping create connections between what students already know and new knowledge they acquire
- Organize knowledge around a subject for quick analysis
- Assess students’ understanding to see where their knowledge needs improvement
- Conduct effective brainstorming around a topic
- Present complex concepts in an easily digestible way
- Organize instructional material for courses or curricula
- Can be used as a basis for discussion among students
- Help identify valid and invalid ideas held by students
- Help promote creative and critical thinking among students
- Can be used as an alternative to traditional note-taking and writing assignments.

As an example, the relation between capacity, skills and competence can be describe in the table below.

*Capacity / Skills / Competence in Industry 4.0 [4-7]*

<table>
<thead>
<tr>
<th>Capacity / Skills / Competence</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced simulation and modelling of virtual plants [4]</td>
<td>Evaluate a large amount of data from different sources, optimize production, and make real-time decisions. Deploy machine data and functionality in the 'cloud' to facilitate data-driven services to monitor and control</td>
</tr>
</tbody>
</table>
production processes.
Optimize industrial systems with simulation (virtual plant).
The real-time nature of the inputs and outputs of Industry 4.0 makes real-time simulation and plant virtualization indispensable in plant operations.

| **Interface between humans and machines [4]** | Program and act with robots, which must be present throughout Industry 4.0. |
| **Augmented reality system [4]** | Generate productivity gains. |
| **3D printing [4]** | Manufacturing parts in one step. |
| **Computational engineering [4]** | Understand programming logic to contribute to the process improvement. |
| **Real-time inventory and logistics system [4]** | Identify and monitor parts and products in the production chain. |
| **Learning factory [5]** | Enable practice-oriented learning in an environment close to industrial reality. |
| **Interrelations among electrical, mechanical, and computer components [5]** | Develop innovative strategies, products, and processes to solve problems of quality. |
| **Statistical methods and data analysis techniques [5]** | |
| **Process knowledge [6],[7]** | Analyse the causes of error and execute quality control of the output products. |
| **Efficiency in decision-making [5]** | Solve complex problems more efficiently and in a short time. |
| **Analytical skills [5]** | Analyse a large amount of data. |
| **Be able to learn in an autonomous way in order to deal with unknown situations and contexts, updating yourself in relation to advances in science and technology (permanent updating) [6]** | Constantly empower (update). Be willing to constantly change / upgrade technology. |
| **Collaboration at a distance [7]** | Acting as a member of a team that meets |
The results of the literature review are highlighted by the set of competencies (knowledge and skills) that must be developed in professional education to accompany the new industrial revolution, as well as the importance of integrating efforts by companies, governments and universities.

Conceptual map [8]

The conceptual map [8] shows that the main competencies needed include skills: (leadership, strategic vision of knowledge, self-organization, giving and receiving feedback, pro-activity, creativity, problem solving, interdisciplinarity, teamwork, collaborative work, initiative, communication, innovation, adaptability, flexibility and self-management) and knowledge of contemporary fields (information and communication technology, algorithms, automation, software development and security, data analysis, general systems theory and sustainable development theory).
References


- Method for representing course structures
  - Technical core/Nontechnical core/Technical electives/Nontechnical electives/Other
  - Credit structures

Based on the information from different sources and universities [1-8], to describe a course from a study programme are needed some general data which comes from the curricula and specific data of the course which have to be specified in each particular case, according to the study programme objectives.

The specific data could be categorised as follows:

- Data about the course identification (name, status, evaluation convenor, etc.);
Total estimated time and number of credits (lecture, seminar, laboratory, project, individual study etc.);
- Prerequisites (if applicable) curriculum-related and competences-related;
- Facilities required for course development and for seminar/ laboratory/ project development;
- Intended learning outcomes of the course (ILOs)/ Specific competences (professional competences and transversal competences);
- Course objectives (general and specific course objectives)
- Topics to be covered/Contents (for lecture/seminar/laboratory/project including teaching-learning methods);
- Evaluation (lecture/seminar/ laboratory/ project);
- Date when the course is certified/ approved by Department/Faculty board.

A possible block scheme of a course structure which contains all the fields categories are presented below.

<table>
<thead>
<tr>
<th>1. Data about the course identification</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Name of course;</td>
</tr>
<tr>
<td>- Study year;</td>
</tr>
<tr>
<td>- Semester;</td>
</tr>
<tr>
<td>- Name of faculty member responsible for the course/ Course convenor;</td>
</tr>
<tr>
<td>- Course status (content);</td>
</tr>
</tbody>
</table>

Depending on the educational system.

For example, in Romania are the following options:

- for the Bachelor level: FC (fundamental course) / DC (course in the study domain)/ SC (speciality course)/ CC (complementary course);
- for the Master level: PC (proficiency course)/ SC (synthesis course)/ AC (advanced course);

| - Course status (attendance type);     |
| - Usually are selected one of the following options: - CPC (compulsory course)/ EC (elective course)/ NCPC (non-compulsory course); |
| - Evaluation type (Exam, Colloquium)  |

<table>
<thead>
<tr>
<th>2. Total estimated time and number of credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Number of hours (total/week), out of which: lecture, seminar, laboratory and project;</td>
</tr>
<tr>
<td>- Study of textbooks, course support, bibliography and notes (total):</td>
</tr>
<tr>
<td>- Additional documentation in libraries, specialized electronic platforms, and field research (total):</td>
</tr>
<tr>
<td>- Preparation of seminars/ laboratories/ projects, homework, papers,</td>
</tr>
</tbody>
</table>
portfolios, and essays (total);
- Tutorial (total);
- Examinations (total);
- Other activities (total);
- Total number of hours (all activities);
- Number of credits (total and per lecture, seminar, laboratory and project).

<table>
<thead>
<tr>
<th>3. Prerequisites (if applicable)</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of courses or other requirements (skills/competences) that are prerequisites enrolling in the course</td>
</tr>
<tr>
<td>- Curriculum-related;</td>
</tr>
<tr>
<td>- Competences-related.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. Facilities required</th>
</tr>
</thead>
<tbody>
<tr>
<td>- For course development;</td>
</tr>
<tr>
<td>- For seminar/ laboratory/ project development.</td>
</tr>
</tbody>
</table>
5. Intended learning outcomes of the course (ILOs)/Specific competences

ILOs depend on level of qualification awarded

- Competences
- Professional competences
- Transversal competences

6. Course objectives

(resulting from the ILOs/specific competences to be acquired)

- General objective
- Specific objectives

7. Topics to be covered/Contents

- Lecture /teaching methods/no. of hours /bibliography
- Seminar / laboratory / project /teaching-learning methods/ no. of hours/ bibliography

8. Evaluation

- Exam/Colloquium
- Course/Seminar/ laboratory/ project

For each is specified: evaluation criteria, evaluation methods and percentage of the final grade.

9. Approvals

- Dates when the course is certified/approved by Department/Faculty board.

**References**


Method for autonomous mapping of course Learning Outcomes (LOs) to POs

While there is no right or wrong way to handle the mapping process, we understood that each of the approaches below might have strengths and weaknesses of their own. The five levels of mapping learning outcomes are as follow [1]:

- PO-Course mapping
- CO-PO mapping
- Assessments - CO mapping
- Syllabus - CO mapping
- Questions - CO mapping

Level 1: PO-COURSE MAPPING

In this first step, the Learning Outcome is mapped with Program outcomes. Program Outcomes which were earlier referred to as Graduate Attributes specify what we expect from a graduate in his/her 4 years of graduation. In line with Bloom’s Taxonomy, this level of mapping lets the Instructor set up threshold value alongside the program learning.

Level 2: CO - PO MAPPING

Aligning program level outcomes with course-level outcomes is the second level of mapping. This level of outcomes mapping focuses on student learning and it allows faculty to create a visual map of a program. It also explores how students are meeting program-level outcomes at the course level.

Level 3: ASSESSMENTS - CO MAPPING

This level of mapping facilitates the alignment of various Assessments with the Courses Outcomes. It represents what students know and will be able to do as a result of the course (at the course level). This level specifies what ways will be used to assess the outcome and what the total student performance on the assessment was (as well as whether the student achieved the desired percentage levels). Lastly, it might specify what the instructor learned from assessing students and how this has altered his/her way of teaching?

Level 4: SYLLABUS - CO MAPPING

This is the main mapping level. It shows how the entire syllabus will be mapped with the Courses in the given course time.

Level 5: QUESTIONS - CO MAPPING

At this level, inquiry learning can be mapped with the courses.
These five levels of mapping articulate how the curriculum and learning goals match together. As a next step, gathering evidence of student learning follows, which involves determining assessment methods and identifying the measures.

Mapping learning outcomes can have benefits. Here is a list of them.

- It identifies how the required courses add to the achievement of program outcomes
- Expands the student achievement in achieving program outcomes
- Works on reflection by revealing gaps in the curriculum and propagates remedial classes
- The course instructors become fully aware of other courses in which students achieve similar outcomes and can plan their syllabi accordingly. This reinforces outcomes, helps skill/knowledge development, and avoids overlap
- Helps to determine the course prerequisites and program requirements based on how the outcomes fit together
- It shows the logic and overall design of a program and captures the relevance of its courses


Method for Classifying programs - Field/Subject area/PG/UG

Generally, the classification of study programs targets two aspects: the content and organization of the programs on one hand [1,2,3], and ranking the programs in terms of quality, performance and quality management [4,5] on the other hand.

According to [1, 2], two main criteria are used to classify a program. They take into account the field of studies (A), and the level of education (B).

A. Classification of programs based on study field

According to the cited sources, 11 main fields are defined:

- 00 – Generic programs and qualifications
- 01 – Education
- 02 – Arts and humanities
- 03 – Social sciences, journalism and information
- 04 – Business, administration and law
- 05 – Natural sciences, mathematics and statistics
- 06 – Information and Communication Technologies
- 07 – Engineering, manufacturing and construction
- 08 – Agriculture, forestry, fisheries and veterinary
Each field is itself divided to several sub-fields, to describe more accurately narrower study areas [3]

B. Classification of programs based on the level of education

Nine education levels are defined:

- ISCED 0: Early childhood education (‘less than primary’ for educational attainment)
- ISCED 1: Primary education
- ISCED 2: Lower secondary education
- ISCED 3: Upper secondary education
- ISCED 4: Post-secondary non-tertiary education
- ISCED 5: Short-cycle tertiary education
- ISCED 6: Bachelor’s or equivalent level
- ISCED 7: Master’s or equivalent level
- ISCED 8: Doctoral or equivalent level

They are grouped in three categories: Low education (levels 0-2), Medium education (levels 3-4), and Higher education (level 5-8). Since significant differences can be found between the education systems of different countries, some so-called proxies for the student have to be specified for each study level, to allow comparison of the programs. These proxies include: the age of students, the duration of studies, the minimum entrance requirements, type of documents (diplomas) issued upon successful completion of studies, type/level of studies the graduate are eligible for, the extent to which the program is oriented to a specific occupation or trade.

Some other criteria can be used to classify education programs, but they are not formalized in ISCED Classification. They are only mentioned to emphasize particular features of the programs: Institutional and structural programs. The criterion takes into account the type of institution that provides education services: school-based programs, enterprise-based programs, and mixed school/enterprise-based programs. This classification covers more dimensions, as follows: service provider, mode of service provision, type of participants, and mode of participation.

In terms of higher education, specific criteria can be applied to classify the programs:

- attendance: full time, part time, distance learning,
- duration of studies: 2, 3, 4, 5 6 years (or 4,6,8,10,12 semesters);
• number of credits: 120, 180, …. Since usually 30 ECTS are allocated per semester, this criterion might not be relevant for programs organized in semesters. However, if a program has another structure in terms of academic year, ECTS can still be a useful criterion.

Within TEAL 2.0, for data handling purpose, an Identity Card might be defined for each program. This should have the following specific fields, one field for a classification criterion:

• program title / acronym;
• country;
• university;
• study field (00 … 10);
• level of education (5 … 8);
• attendance (1 – full time, 2 – part time, 3 distance learning);
• duration of studies (2 …6);

If a code is associated to each criterion, a unique ID could be bound to each program defined within the project. This might be useful for data handling and classification of the programs within the database. For an easy management of the IDs, some rules should be followed in assigning the codes to each criterion, as shown in Table 5.1

Table 5.1

<table>
<thead>
<tr>
<th>#</th>
<th>Criterion/field</th>
<th>Type of field</th>
<th>(Max) length</th>
<th>Possible values</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Acronym</td>
<td>C</td>
<td>5</td>
<td>IN – India</td>
<td>Depends on the full name of the program</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SL – Sri Lanka</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>IT – Italy</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No – Norway</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ro – Romania</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>………</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Country</td>
<td>C</td>
<td>2</td>
<td>IN – India</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SL – Sri Lanka</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>IT – Italy</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No – Norway</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ro – Romania</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>………</td>
<td></td>
</tr>
<tr>
<td></td>
<td>University that delivers the course</td>
<td>N</td>
<td>2</td>
<td>01, 02, 03 ….</td>
<td>To be assigned to each university in the country which use TEAL 2.0</td>
</tr>
<tr>
<td></td>
<td>Study field, according to ISCED</td>
<td>N</td>
<td>2</td>
<td>00, 01, 02, …10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Level of education,</td>
<td>N</td>
<td>2</td>
<td>05,06,07,08</td>
<td></td>
</tr>
</tbody>
</table>
According to the data in table above, the IDs for two programs in UTBV are:

Program: Manufacturing Engineering, Romania, UTBV, Engineering …., Bachelor’s, Full attendance, 4 years. ID; MERO0107060104

Program Advanced Manufacturing Processes Engineering, Romania, UTBV, Engineering …., Master’s, Full attendance, 2 years, ID AMPERO0107070102

Some other criteria of program classification could target the level of recognition of diplomas (national or international), and if within TEAL 2.O, the program has equivalent(s) among the partner universities.

References


Standards and guidelines for the specification of benchmarking statements

Making the educational objectives/outcomes operational is crucial since these are a certification standard in most countries, offer a fundament for selecting appropriate teaching strategies, help the students focus on what is expected from them, offer support for proper evaluation; offer criteria for determining how efficient the whole educational process is.

There are 2 methods to define learning objectives/outcomes: Robert Mager’s Performance-Based Learning Objectives and Gilbert de Landsheere’s 5 step method [1,2]. These can be used to define general, broad and lesson specific objectives/outcomes.

Mager’s approach - Performance-Based Learning Objectives - may be better suited for general and broad objectives.

According to Mager, a learning objective should include the following three components:

- Performance
- Conditions (under which the learner must perform)
- Criteria (by which the performance is evaluated by another; or, in other words, how well the learner must perform)

The Performance defines the Learning Objective “what the learner will be able to perform as a result of some learning experience.” Example: Be able to write a news article. The Conditions of a Learning Objective should „state the conditions, if any, in which the learner must complete the performance.” It is not always needed to add conditions; they should be used to make things clearer and remove ambiguity.

The conditions will tell the learner things like the following:

- What can I use while doing the performance?
- What will be denied to me?
- In which conditions will the performance have to occur?

The Criteria of a Learning Objective

Finally, the third part of a Mager performance-based learning objective is the criterion or criteria: “Having described what you want your students to do, you can increase the communication power of an objective by telling them HOW WELL you want them to be able to do it.”
Here are some examples:

- Identify four out of five product defects on a moving manufacturing line.
- Close ten boxes in a minute.

It may not always be practical to include criteria in a learning objective.

According to Gilbert de Landsheere, a learning objective should include the following five components:

- Who is going to produce the desired behaviour?
- What observable behaviour will be reached?
- What will be the product of this behaviour (the performance)?
- The behaviour has to happen under what conditions?
- How to we decide that the product is satisfactory (criteria)?

Example: The students will be able to build a radio device by choosing the parts according to a blueprint and the device will be able to receive at least 5 radio stations. It can be noticed that, while being quite similar to the 3 components method, this method further details the performance (who, what behavior, what product).

Teachers currently record the LOs in educational engineering projects and didactic strategies in the form of sentences. It is up to the teachers to formulate the LOs and they do it correctly as much as their pedagogical training allows. There is a tendency to compromise the strictness of formulation as the scope of application increases.

**General Learning Outcomes**

Graduates with the exemplifying qualifications, irrespective of registration category or qualification level, must satisfy the following criteria:

| Knowledge and understanding | They must be able to demonstrate their knowledge and understanding of essential facts, concepts, theories and principles of their engineering discipline, and its underpinnings in science and mathematics. They must have an appreciation of the wider multidisciplinary engineering context and its underlying principles.
|                           | They must appreciate the social, environmental, ethical, economic and commercial considerations affecting the exercise of their engineering judgement. |
| Intellectual abilities     | They must be able to apply appropriate quantitative science and engineering tools to the analysis of problems. |
They must be able to demonstrate creative and innovative ability in the synthesis of solutions and in formulating designs. They must be able to comprehend the broad picture and thus work with an appropriate level of detail.

**Practical skills**
They must possess practical engineering skills acquired through, for example, work carried out in laboratories and workshops; in industry through supervised work experience; in individual and group project work; in design work; and in the development and use of computer software in design, analysis and control. Evidence of group working and of participation in a major project is expected. However, individual professional bodies may require particular approaches to this requirement.

**General transferable skills**
They must have developed transferable skills that will be of value in a wide range of situations. These include problem solving, communication, and working with others, as well as the effective use of general IT [information technology] facilities and information retrieval skills. They also include planning self-learning and improving performance, as the foundation for lifelong learning/CPD [continuing professional development].

### Specific Learning Outcomes in Engineering
Graduates from accredited programmes must achieve the following five learning outcomes, defined by broad areas of learning. As set out here, the outcomes apply to accredited programmes at Bachelor (Honours) level leading to CEng registration.

The weighting given to these different broad areas of learning will vary according to the nature and aims of each programme and other requirements of the relevant engineering institution:

- Knowledge and understanding of scientific principles and methodology necessary to underpin their education in their engineering discipline, to enable appreciation of its scientific and engineering context, and to support their understanding of historical, current, and future developments and technologies;
- Knowledge and understanding of mathematical principles necessary to underpin their education in their engineering discipline and to enable them to apply mathematical methods, tools and notations proficiently in the analysis and solution of engineering problems;
- Ability to apply and integrate knowledge and understanding of other engineering disciplines to support study of their own engineering discipline.
| Engineering Analysis                                      | ▪ Understanding of engineering principles and the ability to apply them to analyze key engineering processes;  
                                                                      ▪ Ability to identify, classify and describe the performance of systems and components through the use of analytical methods and modelling techniques;  
                                                                      ▪ Ability to apply quantitative methods and computer software relevant to their engineering discipline, in order to solve engineering problems;  
                                                                      ▪ Understanding of and ability to apply a systems approach to engineering problems. |
|-------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Design                                                      | ▪ Investigate and define a problem and identify constraints including environmental and sustainability limitations, health and safety and risk assessment issues;  
                                                                      ▪ Understand customer and user needs and the importance of considerations such as aesthetics;  
                                                                      ▪ Identify and manage cost drivers;  
                                                                      ▪ Use creativity to establish innovative solutions;  
                                                                      ▪ Ensure fitness for purpose for all aspects of the problem including production, operation, maintenance and disposal;  
                                                                      ▪ Manage the design process and evaluate outcomes. |
| Economic, social, and environmental context                 | ▪ Knowledge and understanding of commercial and economic context of engineering processes;  
                                                                      ▪ Knowledge of management techniques which may be used to achieve engineering objectives within that context;  
                                                                      ▪ Understanding of the requirement for engineering activities to promote sustainable development;  
                                                                      ▪ Awareness of the framework of relevant legal requirements governing engineering activities, including personnel, health, safety, and risk (including environmental risk) issues;  
                                                                      ▪ Understanding of the need for a high level of professional and |
Below we present a UML diagram describing an LO (using the 3-step method from above) as an individual object and its connections to others.
Although we can use UML diagrams to describe the LO as an individual object and its connections to others, there are attributes of the object that are not clearly describable. What we can include as a limited option input (select) are the verbs which connect the objective to the Bloom’s and SOLO taxonomy (these are related only to the performance). The conditions and the criteria are very diverse.

Therefore, it is more flexible and easier to include the LOs as sentences. The teachers are used to doing it this way anyway. A text input should do fine for this.

However, next to the text input, we should include a link which can open a pop-up or a new page where application at operational level is well explained (how the teacher should do it), including the pedagogical aspects mentioned above.
For the GUI, they can be added like this:

For example, if Moodle will be used, they can be added as a new section for a course. It could be done directly in the new distribution and it could be also delivered as a plugin.

References


The basic use cases (scenarios) are as follows:

- **At the offering institution (running an instance)**

**Learner (Student)**
- Profile (complete, update, etc.)
- Programme (search, explore, enrol, drop, etc.)
- Course (search, explore, add, drop, etc.)
- Classroom (search, explore, join, leave, interact, etc.)
- Content (search, explore, add, drop, assess, etc.)
- Feedback (give, view, etc.)

**Classroom Manager (Teacher)**
- Profile (complete, update, etc.)
- Courses (express interest, prepare the classroom, etc.)
- Classroom (create, suggest/populate content, set timeline, offer, stop, etc.)
- Content (search, explore, add, remove, etc.)
- Student (mentor, monitor, interact, assess, etc.)
- Interactive resources such as forums (interact, etc.)
- Assessment (schedule, grade, etc.)
- Feedback and Report (prepare, collect, summarize, act, etc.)

**Programme Creator / Manager**
- Profile (complete, update, etc.)
- Users (create, add, manage, etc.)
- Program (create, update, plan, add/manage users, etc.)
- Courses (create, search, update, etc.)
- Program Objectives/Intended Learning Outcomes (create, update, map, etc.)

- **Central System**

**Content Creator**
- Profile (complete, update, etc.)
- Content (create, update, search, explore, add resources & assessments, etc.)
• Intended Learning Outcomes (map to content, map to assessment, etc.)

*System (Software)*

• Program (rank, version control, track credits, etc.)
• Course (rank, version control, enable/track mappings, track credits, etc.)
• Classroom (rank, version control, activity log, etc.)
• Content (rank, version control, etc.)
• Assessments (track mappings, track and map marks to grades, etc.)

*Systems Admin (Super Admin)*

• Manage the whole system.

*See also Annex I: User Stories.*

*QA and Ranking Engine Requirements*

The functionality of the QA Engine is to rank courses and course content based on:

• their usage
• student performance based on actual level of attainment of Learning Outcomes
• student feedback in the form of ranking

Thus the QA engine will have to be able to store student engagement and the level of attainment of Learning Outcomes (LOs), and then use a suitable strategy to rank the courses and the course content.

The QA and Ranking Engine thus needs to perform the following functionalities:

• Create and manage actions that the system offers
• Monitor and log each action
• Rank courses and contents based on usage, students’ performance and feedback (anonymous)
• Handle version control
• Track mappings of course LOs to course content and assessments
• Track mappings of student attainment of course LOs through performance (assessments)
• Evaluate the effectiveness of classroom design and delivery based on user statistics and LO attainment
• Store, analyse and evaluate feedback provided by students
• Track credit load of Program with contents followed by a student.
VALIDATION OF PLANNED REQUIREMENTS

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
The system consists of the following broad components:

- Programs and program specifications
- Course modules/course descriptors
- Content/learning modules
- LMS
- QA engine.
In order to meet the user requirements, the following specific features will be included in the TEAL2.0 platform:

- User account management
- Access management
- Personalized dashboard
- Collaborative tools and activities - synchronized session scheduling, forums, workshops
- All-in-one calendar
- Convenient file management
- Text editor with LaTeX typesetting capabilities
- Notifications & Track progress
- Multilingual capability
- Interactive HTML5 compatible content creation and view
- Open standards
- Regular security updates
- Detailed reporting and logs
- Embedding external resources
- Multimedia integration
- Marking workflow
- Peer and self-assessment
- Program creation
- Course creation
- Learning content creation - interactive and static
- Course description classification, management and version control
- Learning content classification, management and version control
- Search for courses and create/edit courses
- Search for content and create/edit content
- LO to PO mapping
- Content to LO mapping
- Assessment creation - HTML5 interactive assessments, quiz, essay, and submission types
• Assessments to LO mapping
• Classroom creation with customizable content and customizable lesson planning
• Classroom Plans, schedule
• Customized enrolment (Competence-based, learning-plan based)
• Custom report builder
• Graphical reporting
• Ranking courses, learning content based on user ranking and actual student attainment
• Interface to incorporate software-based third party open learning resources
• Interface to incorporate hardware-based third party open learning resources
• Interface to incorporate other third party open learning resources
• Hierarchical access management
• Custom certificates
• Customized notifications
• Team management
• Record of Prior Learning
• QA Engine
  • Student assessment performance monitoring and tracking of LO attainment
  • Monitoring student attainment of POs
  • Monitoring student interaction with content
  • Monitoring student interaction with teacher
  • Monitoring student interaction with peers
  • Other QA functionalities
DATA FLOW

- **Input data**
  - Course descriptors
  - Course plans
  - Assessment and evaluation
  - Feedback
  - Course ratings
  - Content ratings
  - Suggested course content
  - Registration data
  - Interaction content
  - Exam-related data

- **Transformation of input data**
  - Course descriptors can be used to create programs by the users
  - Users can set up lesson plans
  - Users can edit learning content
  - Usage statistics is being analysed for QA purposes; autonomous ranking process
  - Exam results and evaluation results are being processed in order to allow assessment of student progress

- **Output data**
  - Programs
  - Courses
  - Database with collaboratively developed learning content
  - Interaction content
  - Assessment of student progress; evaluation records of students
  - Summary of evaluation of programs
  - Summary of evaluation/ranking of content.

The data flow is exemplified in the scheme below:

---

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
## User Stories

<table>
<thead>
<tr>
<th>Type of User</th>
<th>User Story</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Holder</td>
<td>As an unregistered student user, I want to be able to register in the system using my Google, Microsoft, or Facebook account, so that I don’t have to manage another account</td>
</tr>
<tr>
<td></td>
<td>As Unregistered teacher, I can request user account from the Program creator/super admin</td>
</tr>
<tr>
<td></td>
<td>As Unregistered teacher, I need to fill teacher application, so that I can become a verified teacher in the system</td>
</tr>
<tr>
<td></td>
<td>As Unregistered Program creator, I can request user account from the super admin, so that I can manage my institution</td>
</tr>
<tr>
<td></td>
<td>As a registered user I want to be able to log in and out of the system using my Google, Microsoft, Facebook, or Moodle account</td>
</tr>
<tr>
<td></td>
<td>As a forgetful user, I can view a password hint, so that I can recall my password</td>
</tr>
<tr>
<td></td>
<td>As a forgetful user, I can request to reset the password, so that I can create a new password</td>
</tr>
<tr>
<td></td>
<td>As an Program creator, I can create teacher/instructor accounts, so that new teachers/instructors can be added to the system under the institution</td>
</tr>
<tr>
<td></td>
<td>As a super admin, I can create user accounts, so that new users can be added to the system</td>
</tr>
<tr>
<td></td>
<td>As a super admin, I can control user accounts, so that new users can be added to the system</td>
</tr>
<tr>
<td></td>
<td>As a super admin, I can review account requests, so that I can accept or decline them</td>
</tr>
<tr>
<td></td>
<td>As a user, I want to be able to access the system through my web browser</td>
</tr>
<tr>
<td></td>
<td>As a user, I want to be able to access frequently asked questions and additional documentation</td>
</tr>
<tr>
<td></td>
<td>As a user, I want to be able to store user configuration settings such as language settings, theme, etc.</td>
</tr>
</tbody>
</table>

<p>| Student | As a student, I want to find programs, so that I can explore program outcomes, structure, future careers, and areas of studying |
| | As a student, I want to read program reviews and ranking from others, so that I can decide which course will best suit me |
| | As a student, I want to register for a program |
| | As a student, I want to leave a program, so that I can get rid of boring stuff |
| | As a student, I want to select courses that are prescribed for my program, so that I can get to know about the course structure, outcomes, and workload |
| | As a student, I want to register for a course, so that I can learn and complete my program |
| | As a student, I want to drop a course |
| | As a student, I want to find classes that most interest me |
| | As a student, I want to enrol for a class that I am able to take online so that I would not need to travel |
| | As a student, I want to leave a class that I selected and select another class instead |
| | As a student, I want to find contents to add to the class, so that I can find the most suitable and popular contents for my course |
| | As a student, I want to read content reviews and ranking from others, so that I can decide which content will best suit me |
| | As a student, I want to add content |
| | As a student, I want to drop content |
| | As a student, I want to create my own lesson plan in consultation with the teacher |
| | As a student, I want to see which programs I have enrolled |
| | As a student, I want to go to registered programs, so that I can see enrolled courses |
| | As a student, I want to see which courses I have enrolled in a program |</p>
<table>
<thead>
<tr>
<th><strong>Teacher (leading a class)</strong></th>
<th><strong>As a student, I want to find programs, so that I can explore programs which I can contribute to</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>As a student, I want to search contents, so that I can see what others offer</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a student, I want to create a classroom, so that I can teach students</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a student, I want to create a timeline for a classroom</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a student, I want to terminate a classroom</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to accept/ reject students who enrol for the classroom</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to suggest contents from the content database to be used in the classroom</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I can moderate content selection by students</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to enable/ disable offering of classes, so that I can get students enrolled or prevent them from enrolling</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to help students to create their own lesson plan</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to make announcements, so that I can inform things to students</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to create discussion forums, so that I can discuss things with students online</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to reply to forums created by students, so that I can answer their questions and address their problems</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I can monitor student progress</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I can view the forum post count of each student, so that I can identify the activity level of this student in the forum</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I can view user statistics</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to see all students’ submissions/ performance</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to see logged information about submissions</strong></td>
</tr>
<tr>
<td></td>
<td><strong>As a teacher, I want to schedule all the assessments according to ILOs &amp; lesson plan</strong></td>
</tr>
</tbody>
</table>

**As a student, I want to see which classes I have enrolled in**

**As a student, I want to see which contents I have enrolled under a class, along with who teaches, number of credits and the description**

**As a student, I want to access contents that I have chosen for a class so that I can study there**

**As a student, I want to see announcements/news, so that I am aware of what is happening**

**As a student, I want to find other peers who have enrolled in the class, so that I can interact with them**

**As a student, I want to interact with peers and teachers in the class**

**As a student, I want to use a forum in the class, so that I can post problems, share my knowledge with other peers and share my suggestion**

**As a student, I want to access all types of resources provided in the content**

**As a student, I want to get notified of upcoming class activities through email, so that I do not miss any of those**

**As a student, I want to be able to complete all the assessments**

**As a student, I want to be able to view the feedback, and assessments from the teacher**

**As a student, I want to give feedback on the content, class, and the system**

**As a student, I want to contact the admin using a help forum, so that I can solve problems occurring in the system**

**As a student, I want to search information in system using search bar, so that I can get what I want easily**

**As a student, I want to see the calendar, so that I am aware of future events**

**As a student, I want to manage notification, so that I can arrange important reminders**

**As a student, I want to manage my profile**
<table>
<thead>
<tr>
<th><strong>Requirements</strong></th>
<th></th>
</tr>
</thead>
</table>
| **Content creator** | As a teacher, I want to grade all the assessments that require manual grading  
As a teacher, I want to be able to give students feedback on their progress  
As a teacher, I want to contact the admin using a help form/chat/email  
As a teacher, I want to interact with students  
As a teacher, I want to obtain feedback from students  
As a teacher, I want to see other teachers’ profiles  
As a teacher, I want to see student profiles  
as a teacher, I want to get student login records, so that I can monitor their participation  
As a teacher, I want to search information in the system using a search bar, so that I can get what I want easily  
As a teacher, I want to add reminders to the calendar  
As a teacher, I want to integrate third party calendars to the system  
As a teacher, I want to manage notification, so that I can arrange important reminders  
As a teacher, I want to generate student reports, so that I can easily evaluate students  
As a teacher, I want to limit the number of students enrolled in a class  
As a teacher, I want to manage my profile  |
| **System** | As a content creator, I want to create contents with its description, outcomes (ILOs), credits, and all information necessary for classification  
As a content creator, I want to search, list and access existing content modules  
As a content creator, I want to update existing contents of a content module, so that I can create a different content module (the system has to take care of version control automatically)  
As a content creator, I want to add resources such as e-notes, videos, demos, virtual labs, etc. to the content module  
As a content creator, I want to be able to create assessments such as quizzes/exams/assignments  |
| **Program creator** | As a system, I want to monitor all activities and log data  
As a system, I want to rank programs/courses/contents based on usage, student performance and feedback, so that I can sort popular and effective ones  
As a system, I want to handle version control of the resources, so that I can provide different versions based on user edits  
As a system, I want to map course ILOs to classroom content ILOs, so that students have flexibility to select appropriate content  
As a system, I want to track ILOs coverage by a student through assessments such as assignments, quizzes etc. and therefore, the overall PO coverage  
As a system, I want to analyze and evaluate feedback, so that I can improve system performance  
As a system, I want to track credit load for programs and courses, so that I can restrict students within a range  |

---

**Co-funded by the Erasmus+ Programme of the European Union**

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.
As a program creator, I want to search, list and access existing courses
As a program creator, I want to update existing courses, so that I can create a different course
As a program creator, want to plan program delivery such as scheduling, assigning teachers to courses, etc.

**Super Admin**
As a super admin, I want to do whatever system can do

<table>
<thead>
<tr>
<th><strong>Key User Type</strong></th>
<th><strong>Tasks</strong></th>
</tr>
</thead>
</table>
| **Teacher**       | Create e-notes  
                   | Video lectures  
                   | Demonstrations  
                   | Activity/assignments/ exams/ quizzes  
                   | Labs  
                   | Resources  
                   | Feedback forums  
                   | Track student involvement  
                   | Create classroom  
                   | Add contents to the classroom  
                   | Add students to the classroom  
                   | Create a course along with ILOs  
                   | Create content along with defined ILOs  
                   | Add chat to contents/courses with scheduling  
                   | See program, courses, contents, teachers, student ranking |
| **Student**       | Register in the system  
                   | Register for a program  
                   | Select courses  
                   | Select contents (based on interaction with a teacher if desired)  
                   | Download resources in a content  
                   | Face for exams/quizzes |
| **Teacher**       | Complete assignments/ activities  
                   | Collaborate with other peers  
                   | Use forums to ask/ answer questions  
                   | Use feedback forums to evaluate course, content, teachers, programs, institutions  
                   | Use chats to interact with peers/ teachers (private)  
                   | Rank courses, contents, classrooms, institutions  
                   | Log in using Google, Microsoft, Facebook account |
| **System**        | Evaluate programs, courses, contents, teachers, students  
                   | Rank programs, courses, contents, teachers  
                   | Store user records  
                   | Exercise version control of resources  
                   | Map program ILOs to courses / contents / classrooms  
                   | Save new program, course, content after changes of ILOs  
                   | Track ILOs coverage by assignments, exams, quizzes, labs for a course  
                   | Analyze and evaluate feedback |