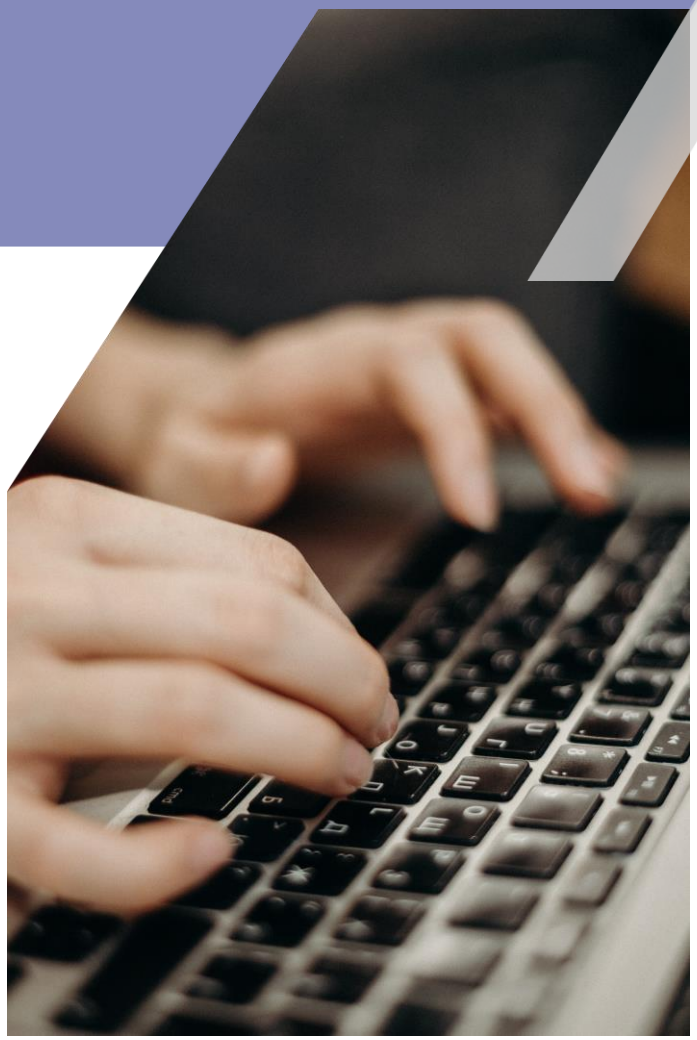
Co-funded by the
Erasmus+ Programme
of the European Union

TEAL2.O
TECHNOLOGY ENABLED ACTIVE LEARNING

TEAL2.O SOFTWARE DESIGN



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

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.

INTRODUCTION

The TEAL2.O Software Design Document translates the software requirements into a representation of the overall architecture, the components, the interfaces and the data necessary in the implementation phase. It spells out more detailed technical specifications and creates a general reference document guiding the next stages of work.

The document should be used in conjunction with the TEAL2.O Analysis model where the requirements are spelled out and the use interface is drafted.

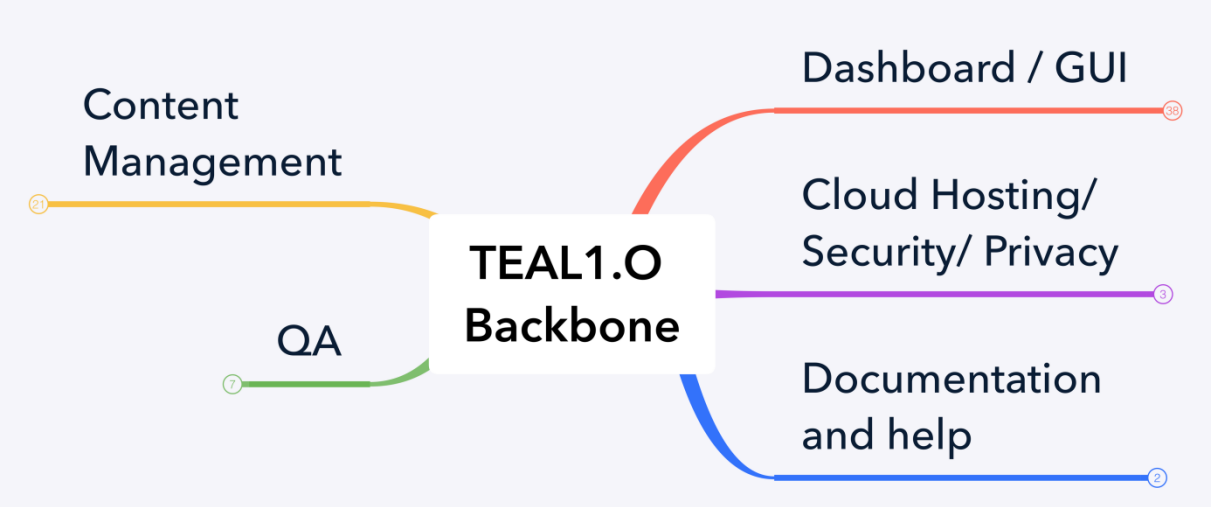
Co-funded by the Erasmus+ Programme of the European Union

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.

FEATURES AND COMPONENTS OF TEAL2.O

The development of TEAL2.O should start by developing the backbone structure that we have named TEAL1.O. This backbone has five main components, each of which has individual "business" value, i.e. serves a separate delineated function while contributing to overall system performance.

TEAL1.O has the following component structure:

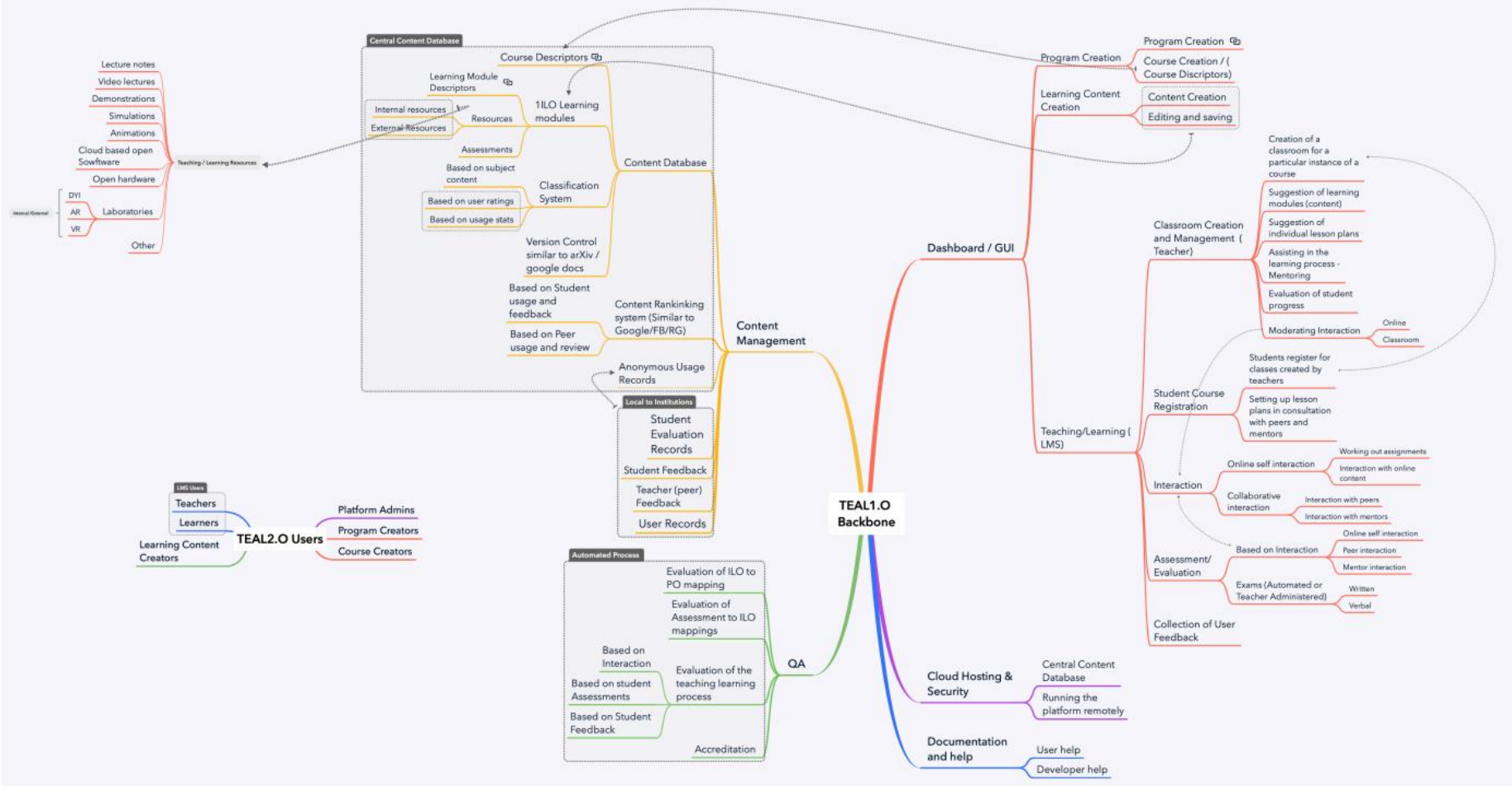


Within the components, there are numerous modules that cannot run as stand-alone functional components but nevertheless allow for decomposition of the development work.

A more detailed graphical representation of the TEAL1.O backbone is presented in the following model:

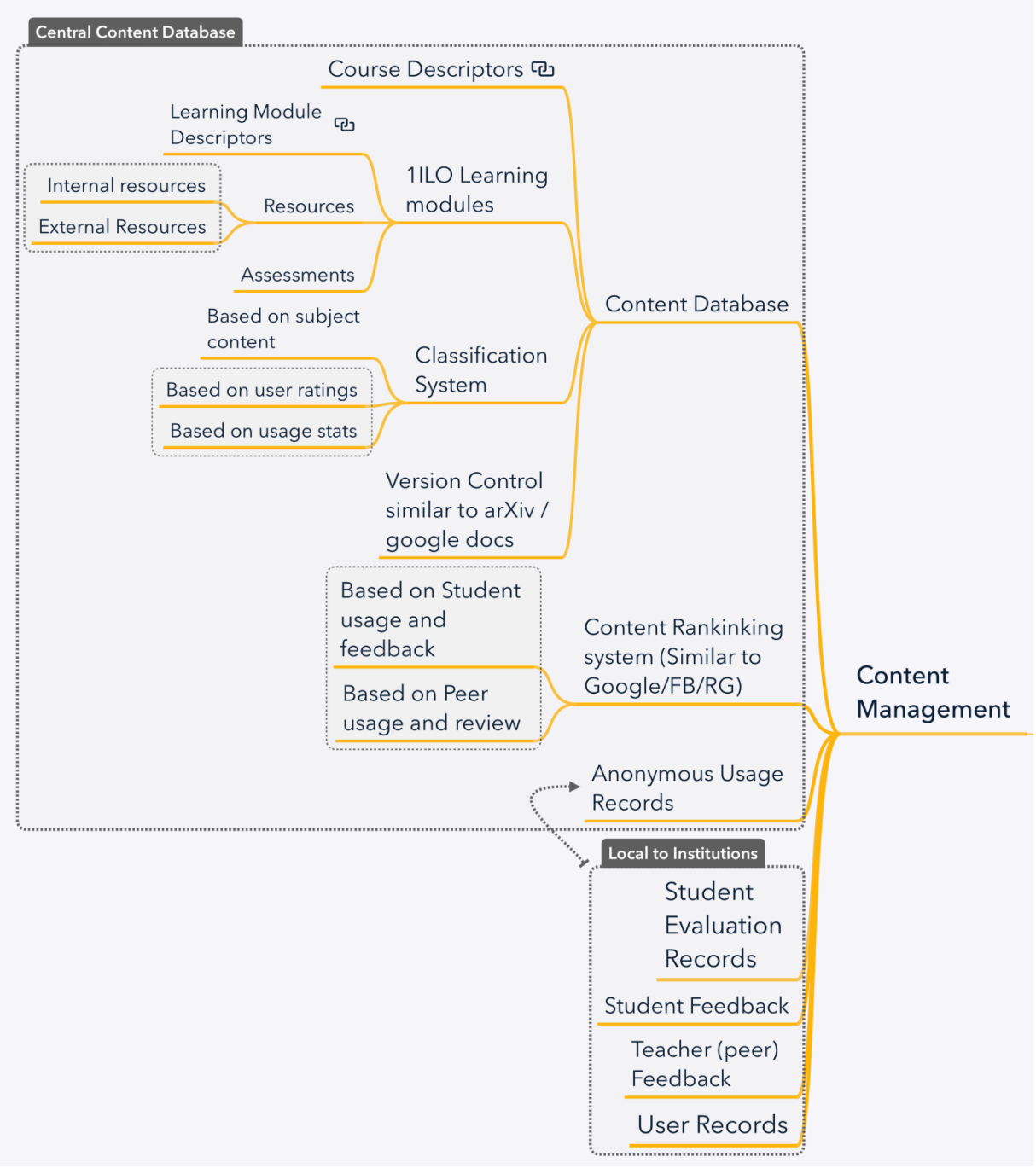
Co-funded by the Erasmus+ Programme of the European Union

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.


TEAL1.O Backbone
Content Management
Central Content Database
Course Descriptors
Learning Module Descriptors
1ILO Learning modules
Resources
Internal resources
External Resources
Assessments
Content Database
Classification System
Based on subject content
Based on user ratings
Based on usage stats
Version Control similar to arXiv / google docs
Content Rankinking system (Similar to Google/FB/RG)
Based on Student usage and feedback
Based on Peer usage and review
Anonymous Usage Records
Local to Institutions
Student Evaluation Records
Student Feedback
Teacher (peer) Feedback
User Records
Teaching / Learning Resources
Lecture notes
Video lectures
Demonstrations
Simulations
Animations
Cloud based open Sowftware
Open hardware
Laboratories
DYI
AR
VR
Other
TEAL2.O Users
LMS Users
Teachers
Learners
Learning Content Creators
Platform Admins
Program Creators
Course Creators
QA
Automated Process
Evaluation of ILO to PO mapping
Evaluation of Assessment to ILO mappings
Evaluation of the teaching learning process
Based on Interaction
Based on student Assessments
Based on Student Feedback
Accreditation
Dashboard / GUI
Program Creation
Program Creation
Course Creation / (Course Discriptors)
Learning Content Creation
Content Creation
Editing and saving
Teaching/Learning (LMS)
Classroom Creation and Management (Teacher)
Creation of a classroom for a particular instance of a course
Suggestion of learning modules (content)
Suggestion of individual lesson plans
Assisting in the learning process - Mentoring
Evaluation of student progress
Moderating Interaction
Online
Classroom
Student Course Registration
Students register for classes created by teachers
Setting up lesson plans in consultation with peers and mentors
Interaction
Online self interaction
Working out assignments
Interaction with online content
Collaborative interaction
Interaction with peers
Interaction with mentors
Assessment/ Evaluation
Based on Interaction
Online self interaction
Peer interaction
Mentor interaction
Exams (Automated or Teacher Administered)
Written
Verbal
Collection of User Feedback
Cloud Hosting & Security
Central Content Database
Running the platform remotely
Documentation and help
User help
Developer help

Here, we present a more detailed view of each component:

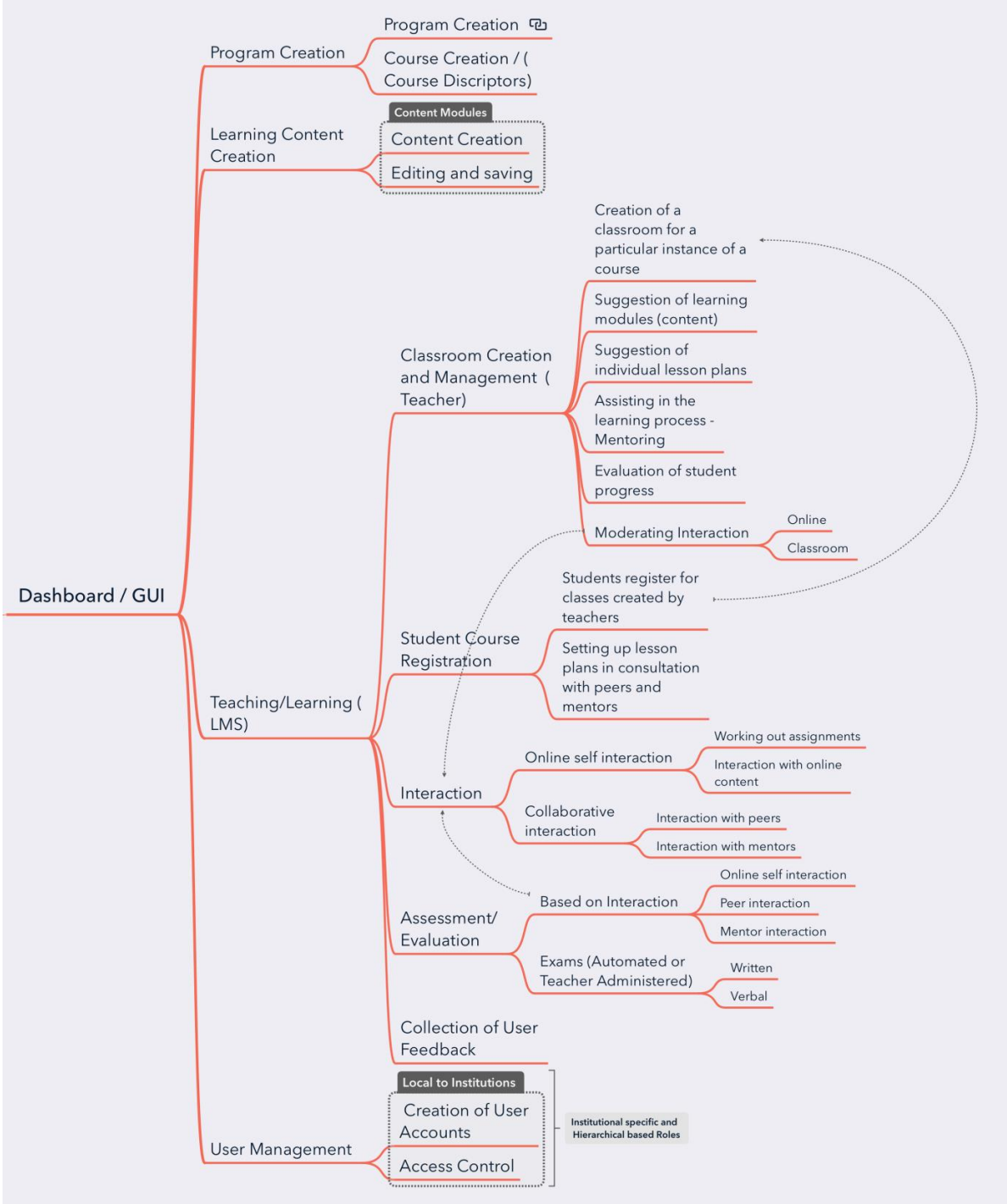
Content Management System



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.

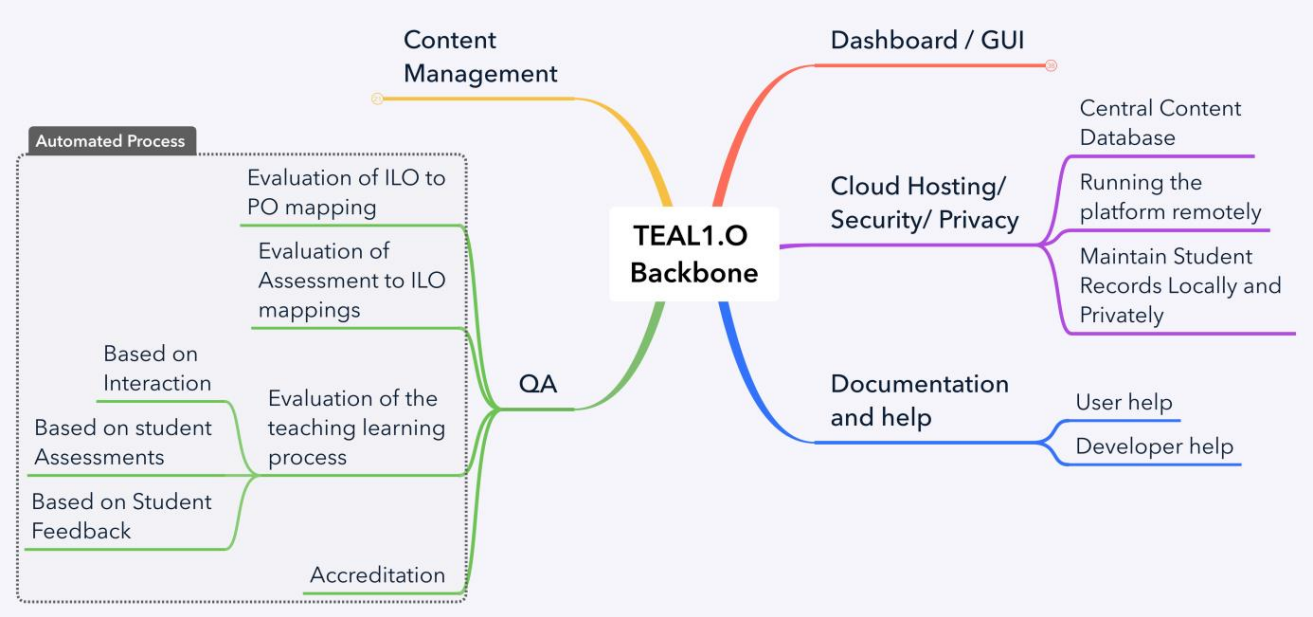
GUI/Dashboard



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.

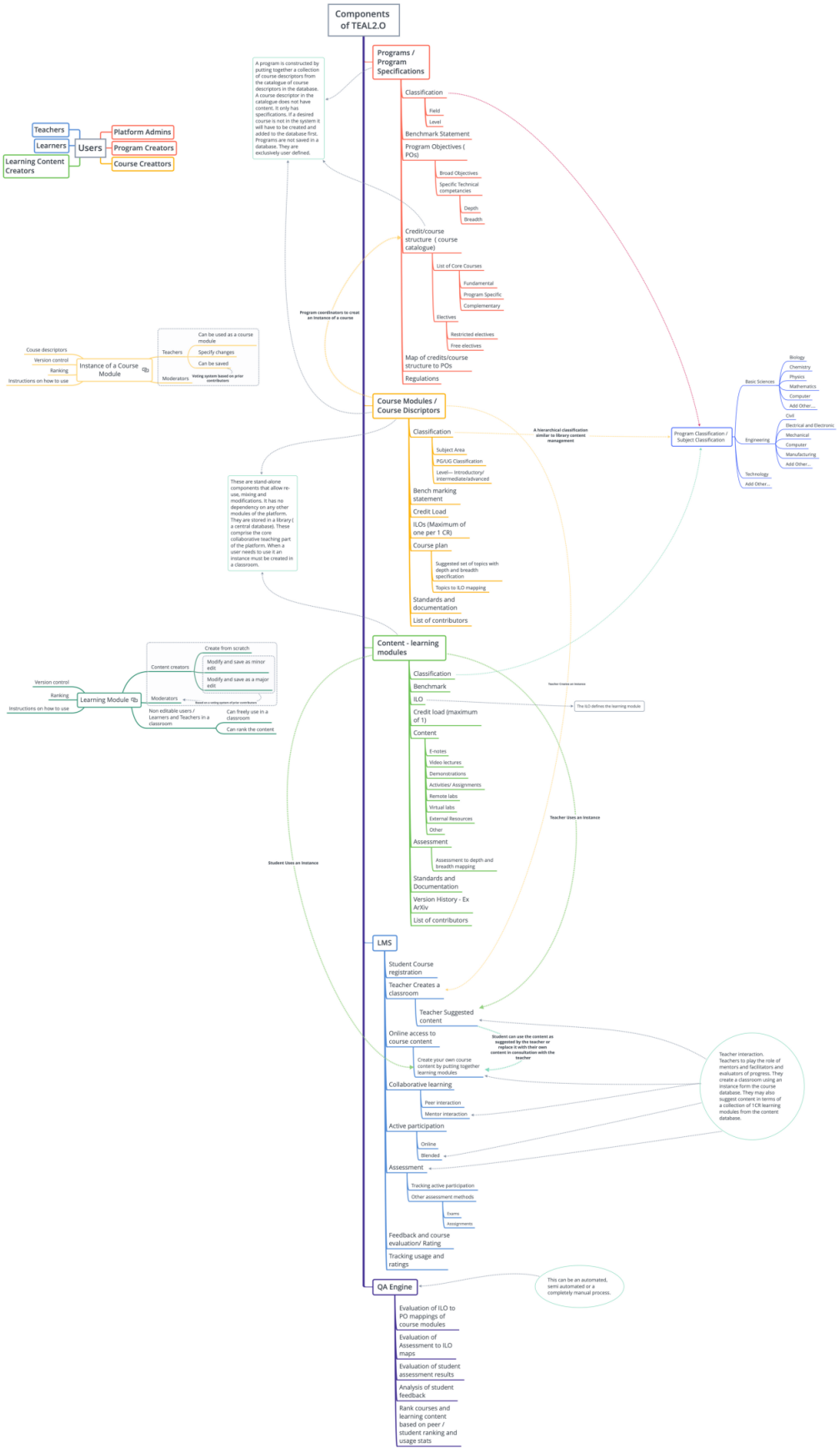
QA and Ranking Engine/ Hosting/ Documentation



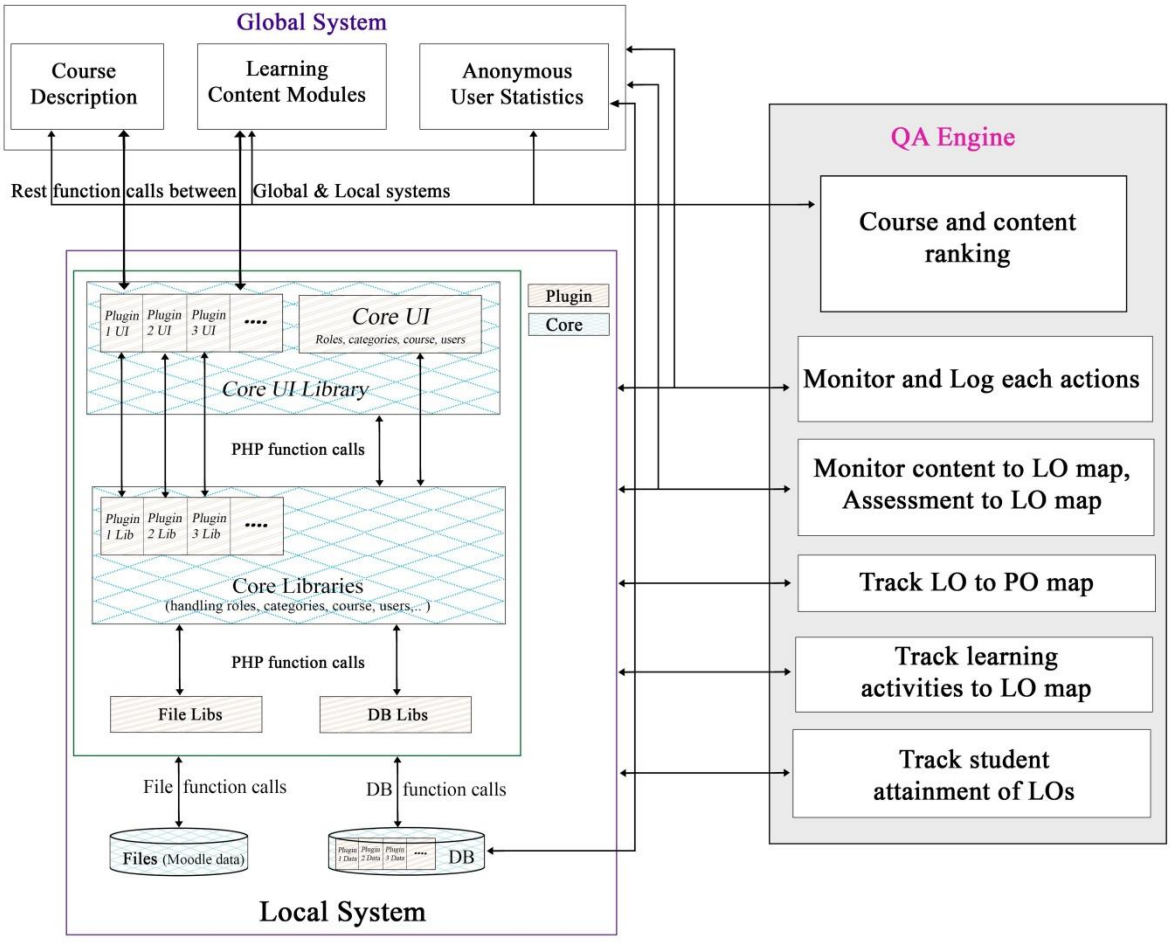
The graph below presents the overall system:

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.


Components of TEAL2.O
Programs / Program Specifications
A program is constructed by putting together a collection of course descriptors from the catalogue of course descriptors in the database. A course descriptor in the catalogue does not have content. It only has specifications, if a desired course is not in the system it will have to be created and added to the database first. Programs are not saved in a database. They are exclusively user defined.
Classification
Field
Level
Benchmark Statement
Program Objectives (POs)
Broad Objectives
Specific Technical competencies
Depth
Breadth
Credit/course structure (course catalogue)
List of Core Courses
Fundamental
Program Specific
Complementary
Electives
Restricted electives
Free electives
Map of credits/course structure to POs
Regulations
Program coordinators to creat an instance of a course
Course Modules / Course Descriptors
Classification
Subject Area
PGr/UG Classification
Level— Introductory/ intermediate/advanced
Bench marking statement
Credit Load
ILOs (Maximum of one per 1 CR)
Course plan
Suggested set of topics with depth and breadth specification
Topics to ILO mapping
Standards and documentation
List of contributors
A hierarchical classification similar to library content management
Program Classification / Subject Classification
Basic Sciences
Biology
Chemistry
Physics
Mathematics
Computer
Add Other....
Engineering
Civil
Electrical and Electronic
Mechanical
Computer
Manufacturing
Add Other...
Technology
Add Other...
These are stand-alone components that allow re-use, mixing and modifications. It has no dependency on any other modules of the platform. They are stored in a library (a central database). These comprise the core collaborative teaching part of the platform. When a user needs to use it an instance must be created in a classroom.
Content - learning modules
Classification
Benchmark
ILO
The ILO defines the learning module
Credit load (maximum of 1)
Content
E-notes
Video lectures
Demonstrations
Activities/ Assignments
Remote labs
Virtual labs
External Resources
Other
Assessment
Assessment to depth and breadth mapping
Standards and Documentation
Version History - Ex ArXiv
List of contributors
Teacher Uses an Instance
Student Uses an Instance
Teacher Creates an Instance
Users
Teachers
Learners
Learning Content Creators
Platform Admins
Program Creators
Course Creattors
Instance of a Course Module
Couse descriptors
Version control
Ranking
Instructions on how to use
Teachers
Can be used as a course module
Specify changes
Can be saved
Moderators
Voting system based on prior contributors
Learning Module
Version control
Ranking
Instructions on how to use
Content creators
Create from scratch
Modify and save as minor edit
Modify and save as a major edit
Moderators
Based on a voting system of prior contributors
Non editable users / Learners and Teachers in a classroom
Can freely use in a classroom
Can rank the content
LMS
Student Course registration
Teacher Creates a classroom
Teacher Suggested content
Online access to course content
Create your own course content by putting together learning modules
Student can use the content as suggested by the teacher or replace it with their own content in consultation with the teacher
Collaborative learning
Peer interaction
Mentor interaction
Active participation
Online
Blended
Assessment
Tracking active participation
Other assessment methods
Exams
Assignments
Feedback and course evaluation/ Rating
Tracking usage and ratings
Teacher interaction. Teachers to play the role of mentors and facilitators and evaluators of progress. They create a classroom using an instance form the course database. They may also suggest content in terms of a collection of TCR learning modules from the content database.
QA Engine
This can be an automated, semi automated or a completely manual process.
Evaluation of ILO to PO mappings of course modules
Evaluation of Assessment to ILO maps
Evaluation of student assessment results
Analysis of student feedback
Rank courses and learning content based on peer / student ranking and usage stats

TEAL1.o ARCHITECTURE



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.

☐ The Overall System

The domain model consists of two major subsystems: the local system and the global system

Local System

A local system is a place, physical or virtual, that allows one to construct an academic program and create and run classrooms using tools available in the Global System.

Global System

The global system has two components

- globally accessible cloud-based database that stores all course specifications, learning content, user records (interaction, level of outcome attainment) and user ranking
- data-driven Quality Assurance Engine that ranks content based on outcome attainment and usage.

☐ Components

Components of the Local System

Program

A program is constructed, for instance, by specifying the broad area of study, the type (UG/PG), the program objectives (POs), number of credits, and a list of courses categorized for example as core, technical electives and non-technical electives. The courses are chosen from a catalogue of courses in the global database. If a desired course is not in the system, it will have to be created and added to the database first. The program should also have a mapping of the course Learning Outcomes (LOs) to the POs. Thus a program is merely a set of specifications. Programs are not saved in a database. They are exclusively user-defined.

Program Objectives (POs)

The POs characterize the attributes of a program graduate. The objectives are brief, clear statements that describe the desired learning outcomes of instruction; i.e., the specific skills, values, and attitudes students should exhibit at the end of the program or after a certain time period following the completion of the program. These can for instance be specified on the basis of certain accreditation criteria such as those developed by the Washington Accord, the Sydney Accord, or the Seoul Accord.

Classroom

This is an instance of a course, where the teacher runs a course. In the classroom, teachers interact with students and students interact with content, teachers, and peers. Students also get evaluated in classrooms. Traditional LMSs typically facilitate this process. In TEAL2.O teachers are expected to

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.

play the role of mentors, facilitators and evaluators of progress. They may also suggest content in terms of a collection of learning modules. Students may also have the freedom to choose the learning modules in collaboration with the teacher. This requirement is not facilitated in a typical LMS.

Specifically, in a classroom teachers can offer a course that is in the system; students can enrol, interact with learning resources made available in the form of course content modules, interact with the teacher, interact with peers and be assessed. It is also highly desirable if the students can decide – in collaboration with teachers – on the content that best suits them to achieve the LOs of the course. A classroom should thus facilitate the collection of student-specific content modules and map the LOs of the content modules to the course LOs, as well as keep track of student interaction (with the content, teacher, and peers), and student assessments.

Components of the Global System

Course

Courses are stand-alone components that allow reuse, mixing and modifications. A certain moderation process may be needed. They have no dependency on any other component of the platform and are stored in a global library (global database) that we will refer to as a "course catalogue".

A course is, much like the program, only a set of specifications. The specifications will include a statement of the broad subject area, the level (PG/UG), the aims and objectives, a benchmark statement, the number of credits, and a list of Learning Outcomes (LOs). The LOs must be mapped to a taxonomy level of choice (such as Blooms or SOLO). It is important to note that a course does not have content such as lecture notes, videos, or assessments.

Course Content (learning resources) Module

Course Contents comprise **the core collaborative teaching part of the platform** and are stored in a global library (global database). When a user needs to use a particular content, an instance must be created in a classroom. The instance can be an edited version of the original version. If so, it must be saved as a new version. A certain moderation process and version control are needed.

The content is characterized by a single learning outcome that is mapped to a taxonomy level of choice (for instance Blooms or SOLO). It will also have a credit weight. To encourage modularity, a course content module will be restricted to between 0.5 and 1 credit hours, where a 1-credit hour is defined according to US college standards.

A course module does have content such as E-notes, Video Lectures, Demonstrations, Activities/ Assignments, Remote Labs, Virtual Labs, External Resources, etc. This also has assessments that are used to evaluate the student attainment of the specified LO of the content.

If the desired content is not in the system, it will have to be created and added to the database first.

Course Intended Learning Outcomes (LOs)

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.

Learning Outcomes are statements that describe significant and essential learning that learners have achieved and can reliably demonstrate at the end of a course or a program. Learning Outcomes identify what the learner will know and be able to do by the end of a course – the essential and enduring knowledge, abilities (skills) and attitudes (values, dispositions) that constitute the integrated learning needed by a graduate of a course or program. The key is that they should be measurable and hence be evaluated.

The learning outcomes approach to education means basing program and curriculum design, content, delivery, and assessment on an analysis of the intended outcome. In this outcomes-based approach to education, the ability to demonstrate learning is the key point.

The structure of a Learning Outcome statement is as follows:

- an action word that identifies the performance to be demonstrated
- a learning statement that specifies what learning will be demonstrated in the performance
- a broad statement of the criterion or standard for acceptable performance.

These outcomes must be mapped to a learning outcome taxonomy of choice such as Blooms or SOLO.

Quality Assurance and Ranking Engine

The functionality of the QA Engine is to rank courses and course content based on their usage and student performance, the actual level of attainment of Learning Outcomes, as well as student feedback. Thus the QA Engine will have to be able to store data on student engagement and on the level of attainment of Learning Outcomes, and to then use a suitable strategy to rank the courses and the course content. The data related to student engagement and student attainment of Learning Outcomes should be anonymously stored in the global database to ensure privacy.

Standards & Documentation

Specifications or other precise criteria are designed to be used consistently, as a rule, guideline, or definition to develop a program. They have to focus on what students will need to learn in order to be competitive on the job market.

Other nomenclature

- **Assessment** – A student evaluation designed specifically to assess the attainment of a given LO
- **Revision** – Store every version of a course and of course content that is created by contributors. The changes in each version should be easily visible and tractable. A certain moderation process may also be needed
- **Credit** – It is yet to be unambiguously defined what one credit of learning engagement will be.

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.

☐ **Roles**

Teacher / Classroom Manager

Creates and maintains a classroom.

Student / Learner

Acts as a student/learner in a classroom

Program Creator

Creates a program (local role)

Course Content Contributor

Creates courses and contents (global)

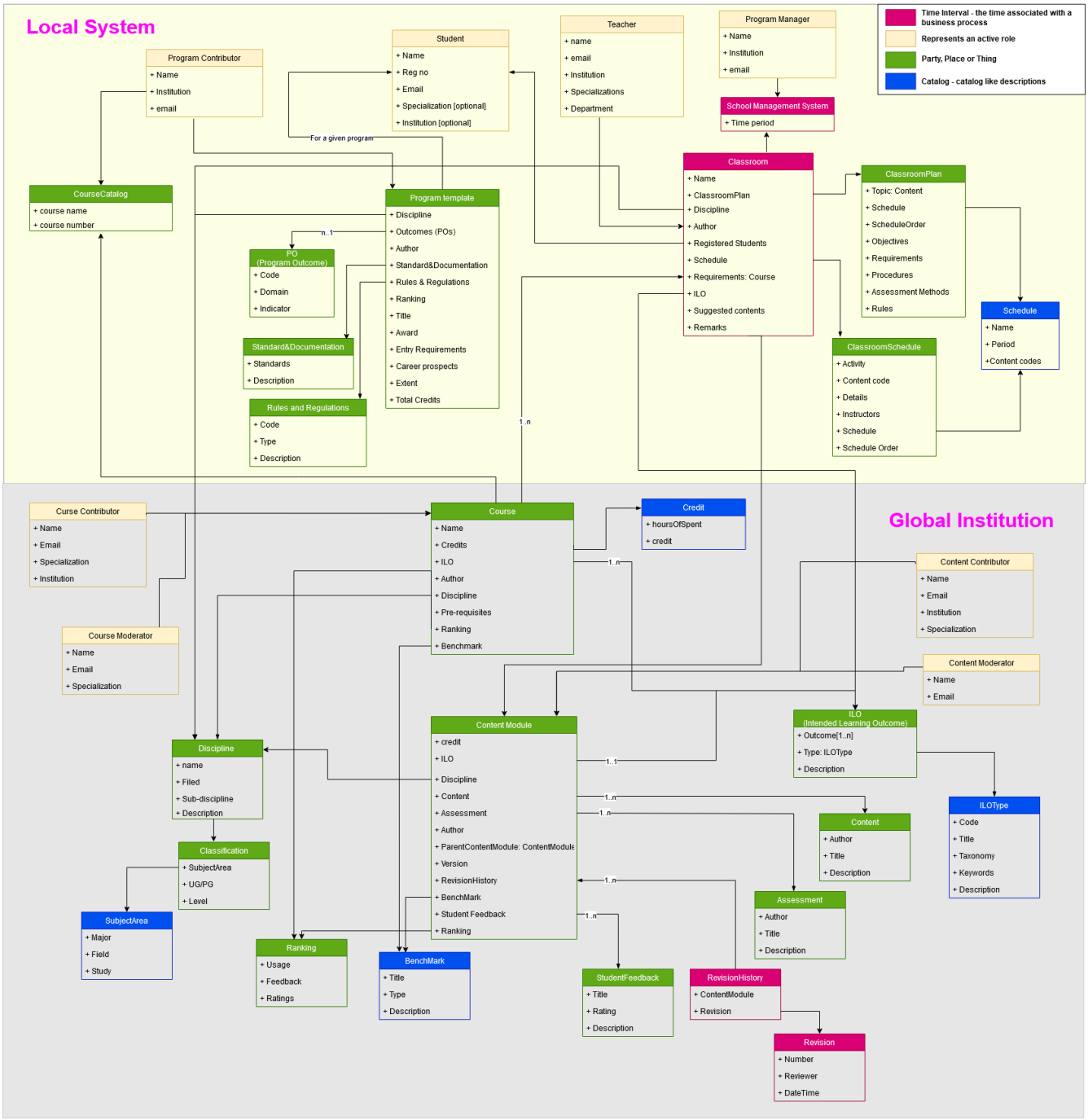
Course Content Moderator

Moderates courses and contents (global).

The domain model is presented graphically below:

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.



Legend

- **Pink**: Time Interval. This category relates to the time associated with a business process or a moment in time. Everything that has time interval or a moment when some event happened comes under this category. For

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.

example, the purchase order for an item can be categorized under pink as it will have the date and time the item was purchased. Therefore, these become tracking details.

- **Yellow**: An Active Role. This category represents an active role that an individual or an organization can play. A person can play a unique role or multiple roles.
- **Green**: Party, Place or Thing. This category can recognize additional attributes like registration number, student's name, etc.
- **Blue**: Catalogue like. In this category, mostly catalogue like descriptions are included. This attribute will have the list of all the details and characteristics for one particular party or activity.

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.

Although Moodle is open source and we can change anything in Moodle to support our needs, the best and most sustainable way to extend it is to write a plugin (sometimes called "a module"). This allows us to incorporate Moodle updates and we will only need to maintain and update the plugins we have developed. It should be reiterated here that the purpose of ensuring Moodle integration is to allow users who already use Moodle as their LMS to incorporate the TEAL2.O features through Moodle in a very convenient way.

Moodle supports a wide range of plugin types. It supports standard and advanced (admin) plugins. The complete list of plugins type details can be found at Plugin types.

Moodle offers a thorough documentation for development. Developer documentation can be found here. It has information on several types of API that are needed to connect with its core and other external systems. These are essential when writing Moodle plugins.

There are several guidelines to follow when developing Moodle plugins. They can be found in dev doc. In Moodle dev doc, they have listed several development tools that are necessary and/or useful. Overview of the communication between Moodle components can be found here.

☐ Standard plugins

Moodle has a general philosophy of modularity. There are nearly 30 different standard types of plugins and even more sub-plugin types. However, all of these plugin types work in the same way. Blocks and activities are the only small exceptions.

See Moodle plugins and Moodle sub-plugins for more information.

☐ Local plugins

The recommended way to add new functionality to Moodle is to create a new standard plugin (module, block, auth, enrol, etc.).The local plugins are mostly suitable for things that do not fit standard plugins.

☐ Custom/ local plugins

- Local plugins are used in cases when no standard plugin fits, examples are:
- event consumers communicating with external systems
- custom definitions of web services and external functions
- applications that extend Moodle at the system level (hub server, amos server, etc.)
- new database tables used in core hacks (discouraged)
- new capability definitions used in core hacks
- custom admin settings

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.

- extending the navigation block with custom menus.

List of differences from normal plugins:

- always executed last during install/upgrade - guaranteed by order of plugins in `get_plugin_types()`
- are expected to use event handlers - events are intended for communication core-->plugins only, local plugins are the best candidates for event handlers
- can add admin settings to any settings page - loaded last when constructing admin tree
- do not need to have any UI - other plugins are usually visible somewhere.

☐ Coding Guidelines

Developer teams should become thoroughly familiar with Moodle's: coding guidelines. These guidelines should be strictly followed.

☐ Tutorial

There is a Tutorial to help you learn how to write plugins for Moodle from start to finish, while showing you how to navigate the most important developer documentation along the way.

☐ Moodle 3.9 Database

Full list of tables in Moodle database can be found HERE.

Moodle Developer Video Tutorial can be accessed from: YouTube channel. It shows clips with guides and tips for the Moodle development.

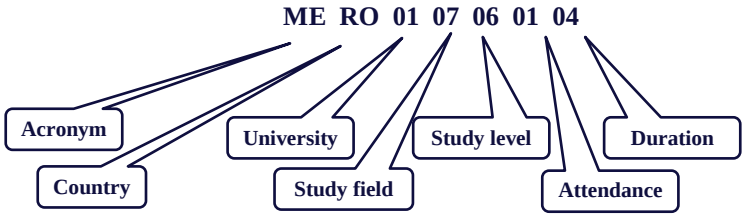
Co-funded by the Erasmus+ Programme of the European Union

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.

☐ **Program Description Tool – Synthesis**

Program		
Field/Parameter	**Data type**	**Description**
1. Data about the program identification		
Title of the program	String	
Acronym of the title	String	
Country the university that hosts the program is located in	Select from strings – see description column	IT – Italy IN – India No – Norway Ro – Romania SL – Sri Lanka …….
University that hosts the program	String	
Study field	Number	According to….. ISCED
Study cycle (level)	Select from strings	Bachelor/Master/Doctoral
Attendance	Select from strings	Full time/Part time/Distance learning
Duration (years)	Number	
Program ID	String	

Optionally, a Program ID could be generated automatically based on the data above, according to the pattern bellow. The structure of the Program ID could be subject of improvement.



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.

Program overview		
Program objectives	Edit window	Defined according to: Washington accord program, Sydney Accord program, ENAEE European guidelines or others.
Intended outcomes	Array of strings	
Competences	Array of strings	
Number of semesters	Number	
Number of weeks/ semester	Number	
Total number of credits (ECTS)	Number	
Supplementary credits (ECTS)	Number	If it applies (e.g. for diploma project / master thesis / …)
Curriculum This data section should be organized in array of records. A record stores data about a course.		
Course	**Record**	
The structure of course record. The names of the fields in record are according to those in ***Table 3 COURSE DESCRIPTION TOOL*** (see next section).		
Course title	String	
Semester	Select from number	1/2
Course status (content)	Select from strings	Depending on the educational system. Example: - 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

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.

		course)/ AC (advanced course);
Course status (attendance type)	Select of strings	CPC (compulsory course) / EC (elective course) / NCPC (non-compulsory course).
Lecture hours / week	Number	
Seminar hours /week	Number	
Laboratory hours / week	Number	
Project hours / week	Number	
Individual study hours (total)	Number	
Number of credits	Number	
Evaluation type	Select of strings	Exam/Colloquium

☐ Course Description Tool – Synthesis

Course		
Field/Parameter	**Data type**	**Description**
1. Data about the course identification		
Name of course	String	
Level	Select from Strings - see description column	(Bachelor, Master, Doctoral)
Study year	Select from Number (1-6)	
Semester	Select from Number (1,2)	
Course status (content)	Select from Strings - see description column	- 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

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.

		course)/ SC (synthesis course)/ AC (advanced course);
Course status (attendance type)	Select of Strings - see description column	CPC (compulsory course) / EC (elective course) / NCPC (non-compulsory course);
Evaluation type	Select of Strings	Exam, Colloquium, aso.
2. Total estimated time and number of credits		
Number of hours (per week)		**Title related to the next 4 fields**
Lecture hours	Number	
Seminar hours	Number	
Laboratory hours	Number	
Project hours	Number	
Individual study hours (total)		Depending on the educational system. Example: Total independent study hours (study of textbooks, course support, bibliography and notes, additional documentation in libraries, specialized electronic platforms, and field research, preparation of seminars/ laboratories/ projects, homework, papers, portfolios and essays, examinations, other activities) is used to have sufficient total hours to justify the number of allocated credits.
Number of credits	Number	Depending on the educational system. For example: Formula (Number of hours + Individual study hours) / 25 = number of credits
3. Prerequisites (if applicable)		
List of the courses or other requirements		Title related to the next 2 fields

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.

(skills/competences) that are prerequisites enrolling in the course		
Curriculum-related	String	
Competences-related	String	
Professional competences	List of selection	A selection (minim 1) from the competences of the program (PC1 to PC6, TC1 to TC3)
4. Required resources		
For course development	String	
For seminar/ laboratory/ project development	String	
5. Intended learning outcomes of the course (ILOs)/Specific competences		
Competences		Title related to the next 2 fields
Professional competences	List of selection	mapped/ from competences of the study program
Transversal competences	List of selection	mapped/ from competences of the study program
6. Course objectives - here we can have Intended learning outcomes of the course (ILOs)		
General objective	String	Related to specific competences
Specific objectives	One Text or multiple String fields added on demand (variant/unknown number)	Related to specific competences - these are ILOs - created with lesson-specific methods - these are strings / sentences created using Robert Mager's Performance-Based Learning Objectives or Gilbert de Landsheere's 5 step method
7. Topics to be covered/Contents		
Lecture /teaching methods/no. of hours	String or table with inputs	Can be text or a table with columns (Lecture name /teaching methods / no. of hours / observations) with a row for each

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.

		lecture/learning unit
Seminar / laboratory / project /teaching-learning methods/ no. of hours	String or table with inputs	Can be text or a table with columns (lesson name /teaching methods / no. of hours / observations) with a row for each seminar/learning unit
Bibliography		1 general field or 2 fields separated for lectures and seminars
8. Evaluation		
Lecture evaluation	String or table with inputs	Can be text or a table with columns (Activity type, evaluation criteria, evaluation methods and percentage of the final grade - these need to be addressed even if the field is one big text) Activity type is "Lecture"
Seminar/ laboratory/ project evaluation	String or table with inputs	Can be text or a table with columns (Activity type, evaluation criteria, evaluation methods and percentage of the final grade - these need to be addressed even if the field is one big text) Activity type is "Seminar/ laboratory/ project"

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.

☐ Mapping Competencies – Synthesis

Professional competencies* Descriptors of structural elements' level of professional competences**	C1 Performing calculations, demonstrations and applications to solve tasks specific to industrial engineering based on knowledge from fundamental sciences.	C2 Associating the knowledge, principles and methods from technical sciences of the field with graphic representations, for solving tasks specific to industrial engineering.	C3 Use of software applications for computer aided design of products.	C4 Elaboration of manufacturing processes	C5 Design and operation of manufacturing equipment	C6 Planning, management and quality assurance of manufacturing processes.
KNOWLEDGE						
1. Knowledge, understanding of basic concepts, theories and methods of the domain and area of specialization; their proper use in professional communication	C1.1 Recognition of important theorems, basic principles and methods specific to fundamental disciplines	C2.1 Identifying the phenomena, theories, and calculation methods specific to the disciplines in the domain and the spatial design of some bodies or their components	.....	.....	.....	.....
2. Using basic knowledge to explain and interpret various types of concepts, situations, processes,	C1.2 Performing demonstrations, explaining and interpreting theoretical results in the	C2.2 Using the knowledge of the disciplines in the field to explain and solve problems	.....	.....	.....	.....

* To be stated at most six professional competences

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.

projects, etc. associated with the domain	use or explanation of theorems or phenomena associated with engineering sciences	and interpret theoretical or experimental results				
SKILLS						
3. Application of basic principles and methods for solving well-defined problems / situations, typical for the domain in conditions of qualified assistance	C1.3 Applying general rules for specific problems of engineering sciences	C2.3 Design of simple parts and subassemblies using appropriately the standards and norms in force	..... .	.....	.....	.....
4. Appropriate use of standard evaluation criteria and methods to assess the quality, merits and limitations of processes, programs, projects, concepts, methods and theories	C1.4 Solving problems of medium complexity and interpreting their results	C2.4 Design of subassemblies and assemblies of medium complexity using appropriately the standards and norms in force	.....	.....	.....	.....
5. Elaboration of professional projects with	C1.5 Choosing the optimal method	C2.5 Achieving projects of	.....	.....	.....	.....

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.

the use of the established principles and methods in the field	and using well-established solutions in solving problems and drawing fair conclusions	medium complexity using computer aided CAD design				
Minimum performance standards for competency assessment:	Standard Identify and explain the basic concepts, principles and methods in the fundamental disciplines. Minimum level: Correctly solving the calculations and mathematical / physical problems of medium complexity, specific to engineering sciences.	Standard Identify and explain the concepts, principles, phenomena, parameters and methods of basic engineering sciences. Minimum level: Correctly solving the of problems of medium complexity, in the domain of industrial engineering sciences, including by using the computing and CAD software	.....	.....	.....	.....
Level descriptors of cross competences**	Cross competences			Minimum performance standards for competency assessment		
6. Responsible execution of professional	.....			.....		

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.

tasks, in conditions of limited autonomy and qualified assistance		
7. Familiarization with the roles and activities specific to teamwork and distribution of tasks for subordinate level	.....	.....
8. Awareness of the need for continuous training; efficient use of learning resources and techniques for personal and professional development	.....	.....

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.

Database Structure

1. ***PROGRAM_CONTRIBUTOR***
    - *PROGRAM_CONTRIBUTOR_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *PROGRAM_CONTRIBUTOR_NAME <VARCHAR2>*
    - *PROGRAM_CONTRIBUTOR_INSTITUTION <VARCHAR2>*
    - *PROGRAM_CONTRIBUTOR_SYSTEM <VARCHAR2>*
    - *PROGRAM_CONTRIBUTOR_EMAIL <VARCHAR2>*

2. ***STUDENT***
    - *STUDENT_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *STUDENT_NAME <VARCHAR2>*
    - *STUDENT_PROGRAM_TEMPLATE_ID <VARCHAR2> (Foreign Key from **PROGRAM_TEMPLATE)***
    - *STUDENT_REG_NO <VARCHAR2> (Unique)*
    - *STUDENT_EMAIL <VARCHAR2>*
    - *STUDENT_SPECIALIZATION <VARCHAR2> (Optional)*
    - *STUDENT_INSTITUTION <VARCHAR2> (Optional)*
    - *STUDENT_SYSTEM <VARCHAR2> (Optional)*

3. ***TEACHER***
    - *TEACHER_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *TEACHER_NAME <VARCHAR2>*
    - *TEACHER_EMAIL <VARCHAR2>*
    - *TEACHER_INSTITUTION <VARCHAR2>*
    - *TEACHER_SYSTEM <VARCHAR2>*
    - *TEACHER_SPECIALIZATION <VARCHAR2>*
    - *TEACHER_DEPARTMENT <VARCHAR2>*

4. ***PROGRAM_MANAGER***
    - *PROGRAM_MANAGER_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *PROGRAM_MANAGER_NAME <VARCHAR2>*
    - *PROGRAM_MANAGER_INSTITUTION <VARCHAR2>*
    - *PROGRAM_MANAGER_SYSTEM <VARCHAR2>*
    - *PROGRAM_MANAGER_EMAIL <VARCHAR2>*

5. ***COURSE_CATALOG***
    - *COURSE_CATALOG_ID <VARCHAR2> (Unique ID - Primary Key)*

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.

- *COURSE_CATALOG_CID <VARCHAR2> (Foreign Key to **COURSE**)*
- *COURSE_CATALOG_PROGRAM_CONTRIBUTOR_ID <VARCHAR2> (Foreign Key to **PROGRAM_CONTRIBUTOR**)*
- *COURSE_CATALOG_NAME <VARCHAR2>*
- *COURSE_CATALOG_NUMERIC <VARCHAR2>*

6. ***PROGRAM_TEMPLATE***
   - *PROGRAM_TEMPLATE_ID <VARCHAR2> (Unique ID - Primary Key)*
   - *PROGRAM_TEMPLATE_PROGRAM_CONTRIBUTOR_ID <VARCHAR2> (Foreign Key to **PROGRAM_CONTRIBUTOR**)*
   - *PROGRAM_TEMPLATE_DISCIPLINE <VARCHAR2>*
   - *PROGRAM_TEMPLATE_OUTCOMES <VARCHAR2> (Foreign Key to the **PROGRAM_OUTCOME**)*
   - *PROGRAM_TEMPLATE_STAND_DOC <VARCHAR2> (Foreign Key to the **STANDARD_DOCUMENTATION**)*
   - *PROGRAM_TEMPLATE_RULES_REGULATIONS <VARCHAR2> (Foreign Key to the **RULES_REGULATIONS**)*
   - *PROGRAM_TEMPLATE_RANKINGS <VARCHAR2>*
   - *PROGRAM_TEMPLATE_TITLE <VARCHAR2>*
   - *PROGRAM_TEMPLATE_AWARD <VARCHAR2>*
   - *PROGRAM_TEMPLATE_ENTITY_REQ <VARCHAR2>*
   - *PROGRAM_TEMPLATE_CAREER_PROSPECTS <VARCHAR2>*
   - *PROGRAM_TEMPLATE_EXTENT <VARCHAR2>*
   - *PROGRAM_TEMPLATE_TOTAL_CREDITS <VARCHAR2>*

7. ***PROGRAM_MANAGER***
   - *PROGRAM_MANAGER_ID <VARCHAR2> (Unique ID - Primary Key)*
   - *PROGRAM_MANAGER_NAME <VARCHAR2>*
   - *PROGRAM_MANAGER_INSTITUTION <VARCHAR2>*
   - *PROGRAM_MANAGER_SYSTEM <VARCHAR2>*
   - *PROGRAM_MANAGER_EMAIL <VARCHAR2>*

8. ***SCHEDULE***
   - *SCHEDULE_ID <VARCHAR2> (Unique ID - Primary Key)*
   - *SCHEDULE_NAME <VARCHAR2>*
   - *SCHEDULE_PERIOD <VARCHAR2>*
   - *SCHEDULE_CONTENT_CODE <VARCHAR2>*

9. ***CLASSROOM_PLAN***
   - *CLASSROOM_PLAN_ID <VARCHAR2> (Unique ID - Primary Key)*
   - *CLASSROOM_PLAN_TOPIC <VARCHAR2>*

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.

- *CLASSROOM_PLAN_SCHEDULE <VARCHAR2> (Foreign Key to **SCHEDULE**)*
- *CLASSROOM_PLAN_SCHEDULE_ORDER <VARCHAR2>*
- *CLASSROOM_PLAN_OBJECTIVES <VARCHAR2>*
- *CLASSROOM_PLAN_REQ <VARCHAR2>*
- *CLASSROOM_PLAN_PROCEDURES <VARCHAR2>*
- *CLASSROOM_PLAN_ASSESSMENT_METHODS <VARCHAR2>*
- *CLASSROOM_PLAN_RULES <VARCHAR2>*

10. ***CLASSROOM_SCHEDULE***

- *CLASSROOM_SCHEDULE_ID <VARCHAR2> (Unique ID - Primary Key)*
- *CLASSROOM_SCHEDULE_ACTIVITY <VARCHAR2>*
- *CLASSROOM_SCHEDULE_CONTENT_CODE <VARCHAR2>*
- *CLASSROOM_SCHEDULE_DETAILS <VARCHAR2>*
- *CLASSROOM_SCHEDULE_INSTRUCTORS <VARCHAR2>*
- *CLASSROOM_SCHEDULE_SCHEDULE <VARCHAR2> (Foreign Key to SCHEDULE)*
- *CLASSROOM_SCHEDULE_SCHEDULE_ORDER <VARCHAR2>*

11. ***CLASSROOM***

- *CLASSROOM_ID <VARCHAR2> (Unique ID - Primary Key)*
- *CLASSROOM_NAME <VARCHAR2>*
- *CLASSROOM_PLAN_ID <VARCHAR2> (Foriegn Key to **CLASSROOM_PLAN)***
- *CLASSROOM_DISCIPLINE_ID <VARCHAR2> (Foriegn Key to **DISCIPLINE)***
- *CLASSROOM_AUTHOR_TEACHER_ID <VARCHAR2> (Foreign Key to **TEACHER)***
- *CLASSROOM_REG_STUDENTS <VARCHAR2> (Foreign Key to **STUDENT)***
- *CLASSROOM_SCHEDULE <VARCHAR2> (Foreign Key to **CLASSROOM_SCHEDULE)***
- *CLASSROOM_COURSE_REQ <VARCHAR2> (Foreign Key to **COURSE)***
- *CLASSROOM_ILO <VARCHAR2> (Foreign Key to **ILO)***
- *CLASSROOM_SUGG_CONT <VARCHAR2>*
- *CLASSROOM_REMARKS <VARCHAR2>*

12. ***STANDARD_DOCUMENTATION***

- *STAND_DOC_ID <VARCHAR2> (Unique ID - Primary Key)*
- *STAND_DOC_STANDARDS <VARCHAR2>*
- *STAND_DOC_DESCRIPTION <VARCHAR2>*

13. ***PROGRAM_OUTCOME***

- *PROGRAM_OUTCOME_ID <VARCHAR2> (Unique ID - Primary Key)*
- *PROGRAM_OUTCOME_CODE <VARCHAR2>*
- *PROGRAM_OUTCOME_DOMAIN <VARCHAR2>*
- *PROGRAM_OUTCOME_INDICATOR <VARCHAR2>*

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.

14. ***RULES_REGULATIONS***
    - *RULES_REGULATIONS_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *RULES_REGULATIONS_CODE <VARCHAR2>*
    - *RULES_REGULATIONS_TYPE <VARCHAR2>*
    - *RULES_REGULATIONS_DESCRIPTION <VARCHAR2>*

15. ***COURSE_CONTRIBUTOR***
    - *CONTRIBUTOR_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *CONTRIBUTOR_NAME <VARCHAR2>*
    - *CONTRIBUTOR_EMAIL<VARCHAR2>*
    - *CONTRIBUTOR_SPECIALIZATION<VARCHAR2>*
    - *CONTRIBUTOR_INSTITUTION<VARCHAR2>*

16. ***COURSE_MODERATOR***
    - *MODERATOR_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *MODERATOR_CONTRIBUTOR_ID <VARCHAR2> (Foreign Key to **COURSE_CONTRIBUTOR**)*
    - *MODERATOR_NAME <VARCHAR2>*
    - *MODERATOR_EMAIL<VARCHAR2>*
    - *MODERATOR_SPECIALIZATION<VARCHAR2>*

17. ***DISCIPLINE***
    - *DISCIPLINE_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *DISCIPLINE_NAME<VARCHAR2>*
    - *DISCIPLINE_FILED<VARCHAR2>*
    - *DISCIPLINE_SUB-DISCIPLINE<VARCHAR2>*
    - *DISCIPLINE_DESCRIPTION<VARCHAR2>*

18. ***CLASSIFICATION***
    - *CLASSIFICATION_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *CLASSIFICATION_DISCIPLINE_ID <VARCHAR2> (Foreign Key to **DISCIPLINE**)*
    - *CLASSIFICATION_SUBJECTAREA<VARCHAR2>*
    - *CLASSIFICATION_UG/PG<VARCHAR2>*
    - *CLASSIFICATION_LEVEL<NUMERIC>*

19. ***SUBJECT_AREA***
    - *SUBJECT_AREA_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *SUBJECT_AREA_CLASSIFICATION_ID <VARCHAR2> (Foreign Key to **CLASSIFICATION**)*
    - *SUBJECT_MAJOR <VARCHAR2>*
    - *SUBJECT_FIELD <VARCHAR2>*
    - *SUBJECT_STUDY <VARCHAR2>*

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.

20. ***RANKING***
    - *RANKING_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *RANKING_USAGE<VARCHAR2>*
    - *RANKING_FEEDBACK<VARCHAR2>*
    - *RANKING_RANKING <VARCHAR2>*

21. ***COURSE***
    - *COURSE_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *COURSE_NAME<VARCHAR2>*
    - *COURSE_CREDITS_ID <VARCHAR2> (Fogreign Key to **CREDIT**)*
    - *COURSE_ILO<NUMERIC> (Foreign Key to **ILO**)*
    - *COURSE_AUTHOR<VARCHAR2>*
    - *COURSE_DISCIPLINE<VARCHAR2> (Foreign Key to **DISCIPLINE**)*
    - *COURSE_PREREQUISIT<VARCHAR2>*
    - *COURSE_RANKING<VARCHAR2>*
    - *COURSE_BENCHMARK<VARCHAR2> (Foreign Key to **BENCHMARK**)*

22. ***CONTENT_MODULE***
    - *CONTENT_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *CONTENT_CREDIT<NUMERIC>*
    - *CONTENT_ILO<VARCHAR2> (Foreign Key to **ILO**)*
    - *CONTENT_CONTRIBUTOR_ID <VARCHAR2> (Foreign Key to **CONTENT_CONTRIBUTOR**)*
    - *CONTENT_MODERTOR_ID <VARCHAR2> (Foreign Key to **CONTENT_MODERATOR**)*
    - *CONTENT_DESICPLINE<VARCHAR2> (Foreign Key to **DISCIPLINE**)*
    - *CONTENT_CONTENT<VARCHAR2> Foreign Key to **CONTENT**)*
    - *CONTENT_ASSESSMENT<VARCHAR2> (Foreign Key to **ASSESSMENT**)*
    - *CONTENT_AUTHOR<VARCHAR2>*
    - *CONTENT_PARENTCONTENTMODULE<VARCHAR2>*
    - *CONTENT_VERSION<VARCHAR2>*
    - *CONTENT_REVISIONHISTOY<VARCHAR2> (Foreign Key to **REVISION_HISTORY**)*
    - *CONTENT_BENCHMARK<VARCHAR2> (Foreign Key to **BENCHMARK**)*
    - *CONTENT_STUDENT_FEEDBACK<VARCHAR2> (Foreign Key to **STUDENT_FEEDBACK**)*
    - *CONTENT_RANKING <NUMERIC> (Foreign Key to **RANKING**)*

23. ***CREDIT***
    - *CREDIT_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *CREDIT_HOURSOFSPENT<VARCHAR2>*
    - *CREDIT_CREDIT<VARCHAR2>*

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.

24. ***STUDENT_FEEDBACK***
    - *FEEDBACK_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *FEEDBACK_TITLE<VARCHAR2>*
    - *FEEDBACK_RATING<VARCHAR2>*
    - *FEEDBACK_DESCRIPTION<VARCHAR2>*

25. ***REVISION_HISTORY***
    - *REVISIONH_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *REVISIONH_CONTENT_MODULE <VARCHAR2>*
    - *REVISIONH_REVISION <VARCHAR2> (Foreign Key to **REVISION**)*

26. ***REVISION***
    - *REVISION_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *REVISION_NUMBER <NUMERIC>*
    - *REVISION_REVIEWER<VARCHAR2>*
    - *REVISION_DATATIME<DATE>*

27. ***ILO***
    - *ILO_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *ILO_OUTCOME <VARCHAR2>*
    - *ILO_ILOTYPE <VARCHAR2>*
    - *ILO_DESCRIPTION <VARCHAR2>*

28. ***CONTENT***
    - *CONTENT_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *CONTENT_AUTHOR<VARCHAR2>*
    - *CONTENT_TITLE<VARCHAR2>*
    - *CONTENT_DESCRIPTION<VARCHAR2>*

29. ***ASSESSMENT***
    - *ASSESSMENT_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *ASSESSMENT_AUTHOR<VARCHAR2>*
    - *ASSESSMENT_TITLE<VARCHAR2>*
    - *ASSESSMENT_DESCRIPTION<VARCHAR2>*

30. ***CONTENT_CONTRIBUTOR***
    - *CONTRIBUTOR_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *CONTRIBUTOR_NAME<VARCHAR2>*
    - *CONTRIBUTOR_EMAIL<VARCHAR2>*
    - *CONTRIBUTOR_INSTITUTION<VARCHAR2>*
    - *CONTRIBUTOR_SPECIALIZATION<VARCHAR2>*

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.

31. ***CONTENT_MODERATOR***
    - *MODERATOR_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *MODERATOR_NAME<VARCHAR2>*
    - *MODERATOR_EMAIL<VARCHAR2>*

32. ***ILO_TYPE***
    - *ILOT_ID <VARCHAR2> (Unique ID - Primary Key)*
    - *ILOT_ILO <VARCHAR2> (Foreign Key to **ILO**)*
    - *ILOT_CODE <NUMERIC>*
    - *ILOT_TITLE<VARCHAR2>*
    - *ILOT_TAXONOMY<VARCHAR2>*
    - *ILOT_KEWORDS<VARCHAR2>*
    - *ILOT_DESCRIPTION<VARCHAR2>*

33. ***USER***
    - *USER_ID <VARCHAR2> (Unique ID - Primary key)*
    - *USER_NAME<VARCHAR2>*
    - *USER_TYPE <VARCHAR2>*
    - *USER_INSTITUTION <VARCHAR2>*
    - *USER_EMAILID <VARCHAR2>*

34. ***PROGRAM_OBJECTIVE***
    - *PROGRAM_ID <VARCHAR2> (Foreign Key)*
    - *PROGRAM_OBJ_DESC <VARCHAR2>*
    - *PROGRAM_ACCRED <VARCHAR2> (Washington Accord, the Sydney Accord, or the Seoul Accord)*

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.