

Appendix I to the Act of Engagement Technical assignment

Online learning and public awareness raising platform of the Central Election Commission of Ukraine

Background information

Within the framework of the Council of Europe Action Plan for Ukraine 2018 - 2021, the Council of Europe is implementing the 2-phased project "Supporting the transparency, inclusiveness and integrity of electoral practice in Ukraine".

Project activities during its Phase 2 (from 1 April 2020 to 31 December 2021) are aimed at:

- (1) enhancing the efficiency and integrity of organization of electoral process and electoral cooperation due to modernized approach and gender mainstreamed election management;
- (2) enhancing public participation and inclusiveness of political and electoral processes;
- (3) increasing women participation in political and public life.

Given constant and frequent changes of the Ukrainian legislation, very often on the eve of respective elections, upon the request of the Central Election Commission of Ukraine (CEC), Project main national partner, the Council of Europe is going to support the development and launch of the permanent CEC online learning and awareness raising platform about elections ("the CEC online Platform").

The CEC online Platform is a joint project of the Central Election Commission of Ukraine and the Council of Europe aimed at providing Ukrainian voters with constantly updated election related information and materials (infographics, leaflets, documents, videos), as well as encouraging them to learn more about elections in Ukraine, electoral process and procedures as foreseen in and by the Ukrainian electoral legislation.

The concept for the CEC online Platform was developed by the Central Election Commission of Ukraine in cooperation and with the support of the Council of Europe local election experts and e-learning consultants.

Partnership: The CEC online Platform will be developed upon the request of the Central Election Commission with the support of the Council of Europe project "Supporting the transparency, inclusiveness and integrity of electoral practice in Ukraine".

Period: The CEC online Platform is planned to be developed by 15 September 2020 on the eve of the 2020 local elections in Ukraine.

Overall objective of the CEC online Platform is to raise the political and legal culture of Ukrainian voters, including their knowledge on issues related to the Ukrainian election legislation, process and procedures. Particularly, it is aimed at developing a modern and inclusive online electoral ecosystem for multiple-format creation, editing and hosting of online courses, educational blogs, games etc. which will be developed by the Central Election Commission of Ukraine with the support of the Council of Europe and other international and national partners of the CEC in the future.

Target group: The CEC online Platform is designed for wide range of audience represented by the following major groups:

1. Electorate. Those who can vote aged 18+

2. Future electorate. Those who are under 18 years and are still obtaining their school or university degree.
3. Subjects of the election process. Those who are directly engaged in the election process (members of election commissions, observers, political parties, candidates, etc).

The CEC online platform with further digital products placed thereon/introduced therein will be in Ukrainian and available for any Ukrainian. For these purposes, the CEC online Platform will be linked to the CEC official website.

Methodology:

The distant platform will enable users to access the CEC online Platform without any time and space limitations.

The navigation between the different e-learning formats within one content block will be conducted with the [Continue] button located under the content block to provide small pieces of information instead of massive chunks of content at once. The dialogue text format should also include interactive blocks that lead to branching so that content of the sections depends on the button that was previously clicked by a user.

The system will have an integrated gamification system that offers achievements on online courses completion represented by the frames around the avatar. The whole avatar achievements is displayed as a popup which appears upon clicking the achievements icon on the top toolbar.

Course creation and editing will happen in the WYSIWYG (what you see is what you get) way, meaning that the course administrator should see everything as it is displayed to the user, basically editing all the course information in inline mode.

Platform Inclusiveness: The online platform shall be fully adapted for people with visual and hearing (production of sub-titles on video products) impairments.

Statistical Data Processing Capacity: The software platform of the course will have technical capacity for data processing (report) that will enable receiving and processing data about the number of enrolled users, the number of users who received certificates, the time needed for the course completion. It would be possible to drill down the information by time (weeks, days, hour). Processed statistical data (protection of personal data is ensured) will be used for program evaluation purposes.

Anticipated outcome: the CEC online Platform is developed and fully operable by 15 September, respective digital products (interactive online courses developed by the CEC and its partners) can be integrated into and run through the Platform.

Technical assignment for web-designing, online platform development and an interactive digital blog instalment and integration is provided below.

Technical Assignment for the CEC online Platform

A) Web design and development of the CEC online Platform

The provider shall develop a core landing page, two subpages of the online courses according to the brand identity and visual guidelines for the Platform, and detailed mock-ups to be provided by the Council of Europe.

Developed landing pages shall be inclusive and adapted to the needs of people with visual impairments.

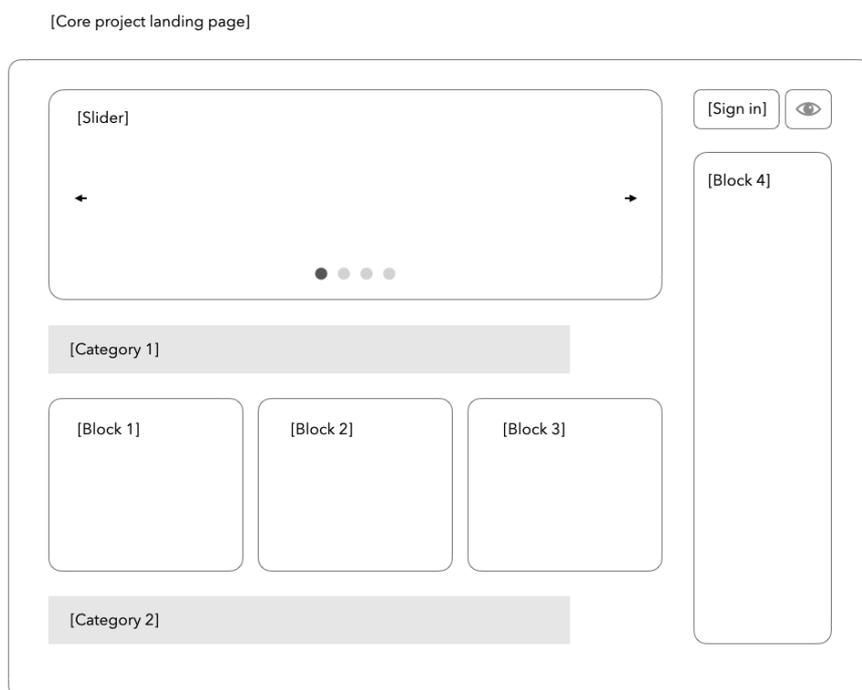
Developed landing pages shall be responsive and suitable for phones, tablets and personal computers/laptops.

The core landing page will interlink and unite online learning objects (to be integrated into the Platform), while the course landing page will provide the user with additional information about a course and its pros. The provider shall develop a core landing page for the Platform, two subpages of the online courses according to the brand identity and visual design of the Platform, and detailed mock-ups to be provided by the Council of Europe.

The core landing page will interlink and unite e-learning objects, while the course landing page will provide the user with additional information about the course and its pros.

Platform core landing page prototype (can be changed during the mock-up development):

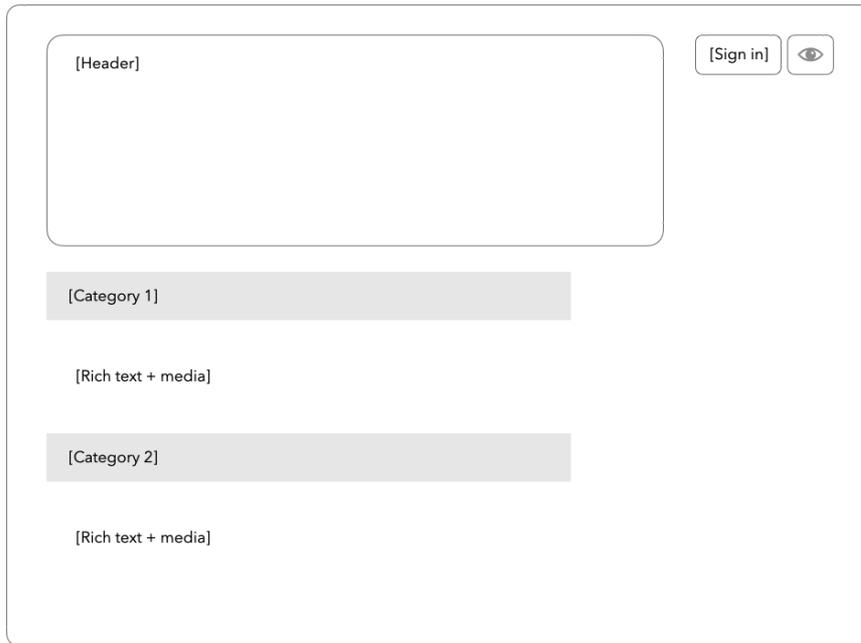
(prototype 1: core landing page prototype)



Online course landing page prototype (can be changed during the mock-up development):

(prototype 2: online course landing page prototype)

[Online course landing page]



B) Adjustment of existing e-learning management system (LMS, either open source or closed source system) to be integrated into the CEC online Platform

A learning management system (LMS) for the purposes of this assignment shall be proposed by the Provider in its technical proposal and shall be either open source or closed source system. In case of the closed source system proposed, the Provider shall submit in its technical proposal all source code(s) for the system. In either case, the Provider shall indicate in its technical proposal the estimated costs for the LMS maintenance, placement, hosting and other relevant costs to be incurred for further good functioning of the system.

The proposed e-learning management system (either open sourced (Moodle or alike) or closed sourced) shall be adjusted according to the needs of the learning modules/objects to be placed on the CEC online Platform and designed according to the brand identity and visual guidelines for the Platform to be provided by the Council of Europe. In case of the closed source LMS adjustment, the Provider shall additionally provide in its technical proposal the source code (zip file only) for the purpose of assessment of the technical proposal. Upon completion (adjustment) of the learning management system, all and any application's "source" code shall be delivered to the Council of Europe and to the Central Election Commission of Ukraine as recipient of the system. This shall include but not be limited to the delivery of all the media (video, audio, pictures and text) as well as the source code for developing the courseware.

User registration page

Main goals of the user registration page are to ensure:

1. User's entry into the System;
2. User's registration in the System;
3. Resetting a user's password.

Requirements for the System:

To meet the main goals of the User's First Page, the following should be included on the first page:

1. Signing in the System;
2. Registration;
3. Reset password.

1) Signing in the System

To log into the System, a User shall fill out two mandatory fields.

1. Field title [E-mail]
Field properties: mandatory; text; Roman type shall be set as default on placing the cursor in the field.
2. Field title [Password]
Field properties: mandatory; numbers only can be entered.

There should be a button [Sign in] below the fields on the form.

On the click of the button [Sign in] the System shall check whether all mandatory fields have been filled out. If any of them has not been completed, a caption "Mandatory" shall appear in red inside the field. If all mandatory fields have been filled out, the System shall check whether the information in the fields has been entered correctly.

If the provided information is incorrect, the message with the text "Wrong credentials" shall appear. If the information has been entered correctly, the main page of the System shall be opened.

Alternatively, a user can sign in the System via Facebook and Google authentication by pressing the [Continue with Facebook] or [Continue with Google] buttons. The buttons should be located below the registration block.

The User will be redirected to the Facebook or Google authentication page for further verification and automatically redirected back to the e-learning platform for further interactions.

(prototype 3: signing in the system)

[Sign in block]

2) Registration in the System

On the First Page, below the button [Sign in] there should be the button [I don't have an account yet]. On the click of the button [I don't have an account yet], the registration form shall open.

Registration form

The registration form shall contain the following fields and buttons (prototype 4):

1. Field title: [Name]
Field properties: mandatory, text. Georgian font Sylfaen shall be set as default on placing a cursor in the field.
2. Field title [E-mail]
Field properties: mandatory; text; Roman type shall be set as default on placing the cursor in the field.
3. Field title [Mobile number]
Field properties: mandatory; numbers only can be entered;
4. Field title [Password]
Field properties: mandatory; text; any character can be entered.
5. Field title: [Register]

Alternatively, a user can sign in the System via Facebook and Google authentication by pressing the [Continue with Facebook] or [Continue with Google] buttons. The buttons should be located below the registration block.

The User will be redirected to the Facebook or Google authentication page for further verification and automatically redirected back to the e-learning platform for further interactions.

Field parameters: mandatory [Name], [E-mail], [Mobile number] and [Password] to be filled out.

Case 1: If User selected/filled out [E-mail], the form should display the field:

- o Field title: [E-mail]

Field properties: mandatory, field for manual entry, Roman type shall be set as default once a cursor is placed in the field. If the entered information is not in mail format, "E-mail must be valid" shall appear below the relevant field.

(prototype 4: registration form)

[Registration form]

The image shows a registration form prototype within a rounded rectangular container. At the top left, the text "[Registration form]" is displayed. Below it are four input fields, each with a label and a grey rectangular placeholder: "[User name*]", "[E-mail*]", "[Mobile number*]", and "[Password*]". Below the fields is a rounded rectangular button labeled "[Register]". Underneath the button is a link labeled "[Forgot password?]", which is underlined. A horizontal line with the text "OR" in the center separates this from two more rounded rectangular buttons: "[Continue with Facebook]" and "[Continue with Google]". At the bottom of the form, there is a link labeled "[I don't have an account yet]", which is underlined.

3) Resetting password in the System

Below the button [Registration] on the first page there should be a button [Forgot password] that will lead to the password reset form (prototype 5). The form shall contain the [E-mail] field and the [Reset password] button.

The following message shall be displayed "Enter the E-mail and click the button [Reset password]". The link will be sent to the entered E-mail".

The following shall be placed below the message:

- o Field title [E-mail]

Field properties: mandatory, text. Roman type shall be set as default on placing a cursor in the field. If the entered information is not of the e-mail format, the message "E-mail must be valid" shall appear. The button [Reset password] shall be placed below the field.

Upon clicking on [Reset password], the System shall check whether the e-mail address is registered in the System. If not, the message "Entered e-mail address is not registered in the System" shall appear.

In case E-mail is registered, the message shall be sent: "To reset the System entry password, please click the link". Upon pressing on the [link] the System shall open the password reset form. The form shall include:

Field title [New password]

Field parameters: mandatory, manual entry. Restriction: Password must have at least 8 characters and should not exceed 16 characters.

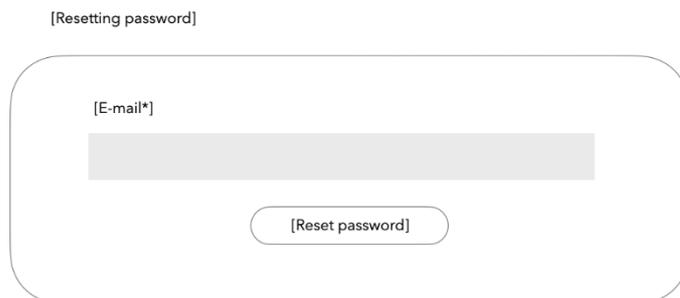
Field title [Re-enter password]

Field parameters: mandatory, manual entry. The System shall check whether it coincides with the symbols entered in the [Password] field. If it does not match, the following message in red color shall appear "Passwords must match".

Below the fields, the button [Reset password] shall be placed. Upon clicking on it the System shall reset the password and display the message: "Password changed successfully."

(prototype 5: password reset form)

[Resetting password]



Gamification

The system has an integrated gamification system that offers achievements on courses completion. It looks like a list of avatars that are disabled until the user hasn't completed the courses. Once the user starts completing courses the avatars are unlocked and the user can choose one of them, so their user picture is replaced. The user can also choose the frames around the avatar and the caption. The whole avatar achievements are displayed as a popup which appears upon clicking the achievements icon on the top toolbar.

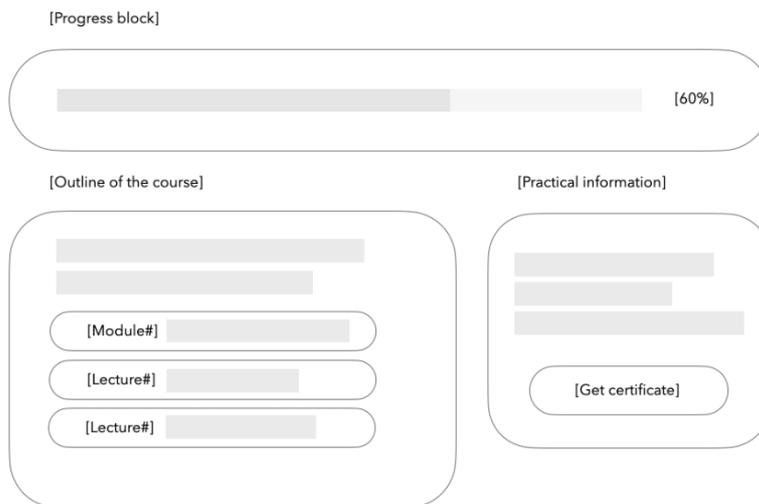
Course Page

Course page consists of 3 main blocks (prototype 6):

1. Progress block. Visual progress bar displaying the number of concluded lectures in the course as well as text displaying the progress in percents
2. Outline of the course. This is a block displaying the list of modules and their lectures. For every module it displays its [title], [description], and, if applicable, [start date]. For every lecture, it displays the [icon] visualizing the type of the lecture (text, video, or mixed), [lecture's title], and [description]. Clicking on the lecture will open the lecture page.
3. Practical information of the course. This block is displayed on the right-hand side and it shows the practical information such as the [number of lectures in the course], the [date the course was started], [total number of questions] and [number of answered questions]. If the course has a certificate, there is also a button that says [receive certificate]. This button is inactive until the learner hasn't answered the questions needed for getting the certificate. Once this condition is accomplished, the button becomes active.

The course page might also contain a graphical element (for example, a lock) which means that the enrolment on the course is blocked until some conditions are met, for example, the completion of another course. The banner contains the information of the needed prerequisite condition with a hyperlink to the course or set of courses (or tests) needed to be done.

(prototype 6: course page)



Lecture page

Lecture page is the smallest e-learning object and consists of two main blocks (prototype 7):

1. Content block
2. Navigation panel

The course content block should support different combinations of the following formats:

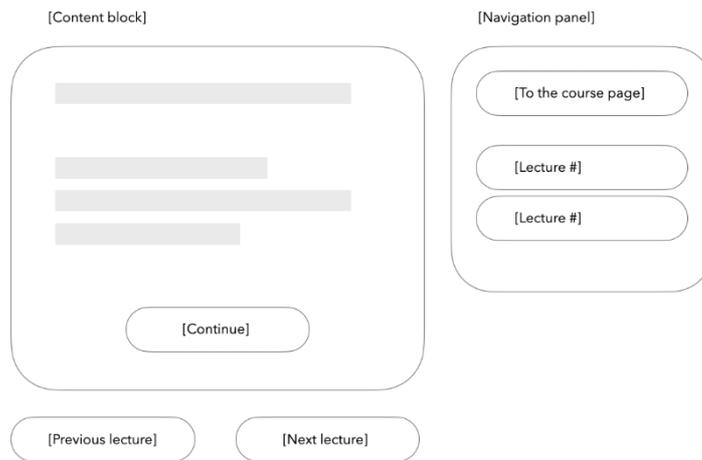
- text format (rich text/html/pdf)
- text & media format (rich text/html/pdf + YouTube/Vimeo videos integration)
- dialogue text format (messenger simulation with rich text/html/pdf/YouTube/Vimeo videos integration).
- single/multiple choice tests

The dialogue text format should also include interactive blocks that lead to branching: the user should be able to select how they want the lecture to proceed. It should look as a list of buttons with specified paths and once the user clicks on them the next sections are open and the content of the sections depends on the button that was clicked (prototype 8).

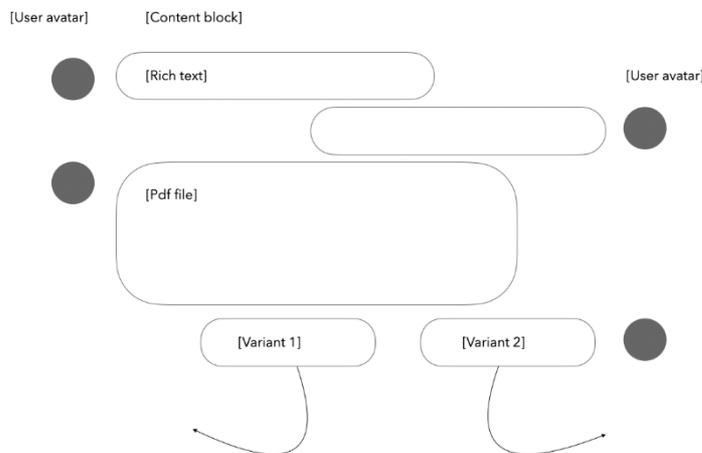
The navigation between the different formats within one content block should be conducted with the [Continue] button located under the content block. This will allow to reveal the course content successively by pressing the [Continue] button to provide small pieces of information instead of massive chunks of content.

The navigation between the different content blocks within one online course should be conducted with the [Previous lecture] and [Next lecture] buttons located at the bottom of the lecture page or be the [Navigation panel] located near the content block.

(prototype 7: lecture page)



(prototype 8: dialogue text format)



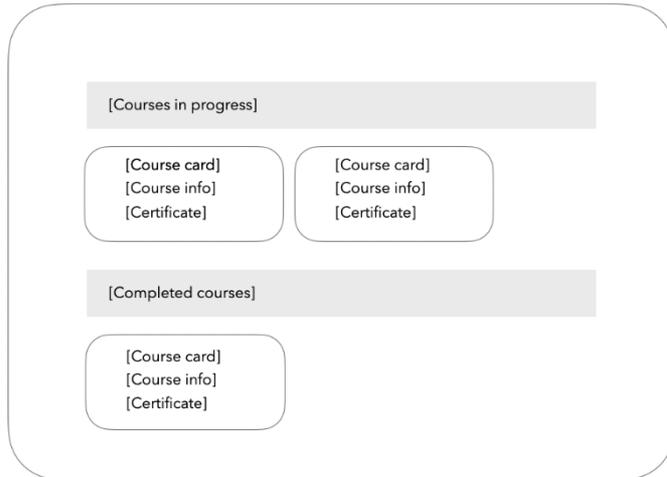
3. User's dashboard. This is a page where the users can see essentially two lists of courses: the [courses in progress] and [completed courses] (prototype 9).

The [courses in progress] are displayed as a list of course cards that are hyperlinked with the corresponding course page. Once the user clicks on the card, they are immediately redirected to the course page.

Completed courses are also displayed as a list of cards and besides the [course image], [title], and [description] as well as they contain a [link to the certificate]. The user can click this button any time and as many times as they want and every time the page with their certificate for the corresponding course will open.

(prototype 9: user's dashboard)

[Users dashboard]

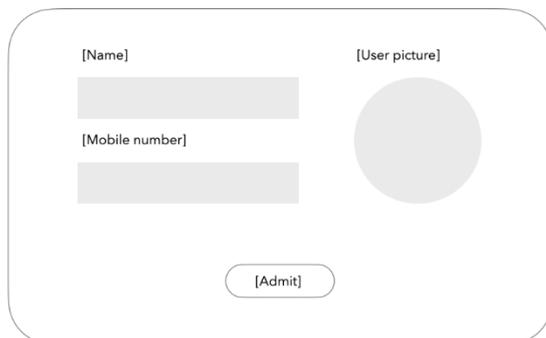


4. Profile Settings

The user should be able to open the profile settings area where they can change their [name], [user picture] and [mobile number] (prototype 10).

(prototype 10: profile settings)

[Profile Settings]



5. Admin Area

Admin users should be able to see a navigation button "Manage courses" that leads to the admin interface where it is possible to create a new course or edit the existing one.

a. *Create a new course* – should be a user-friendly wizard that appears in the popup, guides users step by step containing overall 7 steps (prototype 11):

- i. course title
- ii. course description – should support rich text format, HTML, images, links
- iii. promo-video – should support links to Youtube and Vimeo
- iv. the length of the course – should be in minutes and the system should translate into hours/days/weeks if necessary
- v. indicate whether or not the course will have certificate in the end
- vi. indicate whether or not the course should have a discussion forum
- vii. indicate the basic structure (skeleton) of the course – the number of modules, lectures, and sections (with possibility of changing in the future)

The navigation between steps of the guide should be happening using buttons cancel and continue.

(prototype 11: create a new course)

[Admin Area]

[Course title] [Course description] [Promo-video] [Duration] [Certificate] [Forum] [Structure]

[Course title]

[Cancel] [Continue]

b. *Course editing* (prototype 12):

Editing should happen in the WYSIWYG (what you see is what you get) way, meaning that the course administrator should see everything as it is displayed to the user, basically editing all the course information in inline mode. The course editor should allow to create and edit the following blocks:

- text format (rich text/html/pdf)
- text & media format (rich text/html/pdf + YouTube/Vimeo videos integration)
- dialogue text format (messenger simulation with rich text/html/pdf/YouTube/Vimeo videos integration).
- single/multiple choice tests

There should be an editing toolbar on top that allows text formatting and editing (colours, bold, italics etc), add picture or hyperlink. This toolbar should also contain a button that opens an [HTML formatter popup] allowing creating more complex structures.

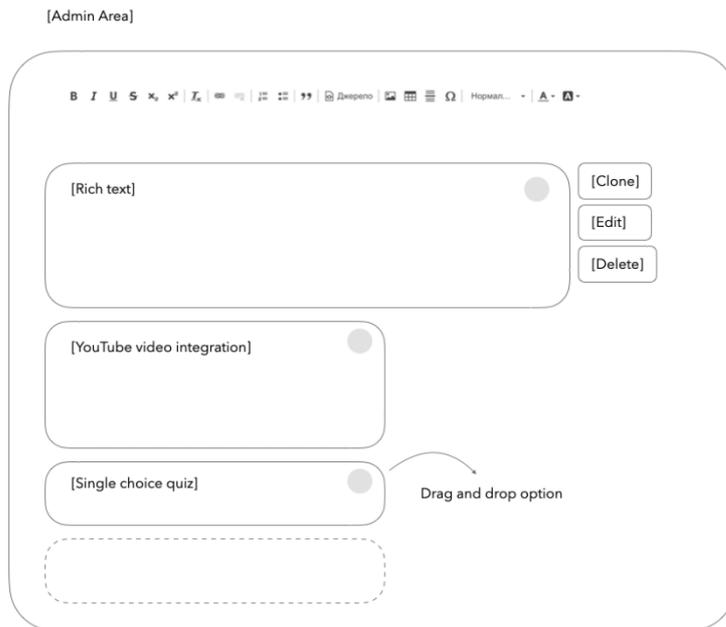
Besides editing course title, description and course basic information, the course editor should allow adding new modules and lectures, use drag and drop to change the course structure and delete the structure elements if necessary.

There should also be a functionality of cloning lectures, so the administrators are able to create lecture templates for easy content management. This means that each lecture item element should contain icons "delete", "clone", and drag and drop icon that allows changing lectures order. Module items elements should also contain buttons "delete" and drag and drop icon.

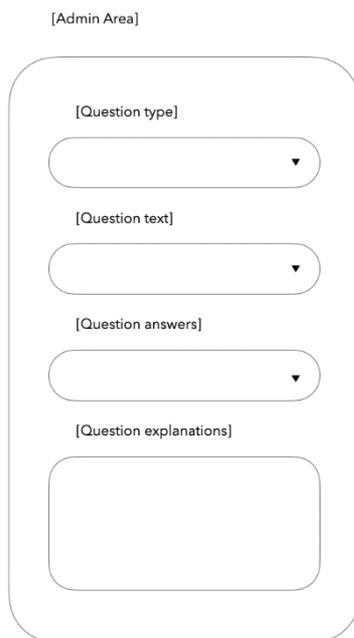
The course editor should allow to create single and multiple-choice quizzes in four steps (prototype 13):

- Question type
- Questions text editor (rich text)
- Answers text editor (rich text)
- Question explanations editor (rich text)

(prototype 12: editing existing course)



(prototype 13: test editor)



C) Digital blog

The provider should install and integrate an interactive digital blog for information storing and sharing. It will combine additional materials from different stakeholders and should consist of (prototype 14,15):

1. Editing (admin) interface allowing editing and adding news/blogs/docs/links.
2. Search option to ease the search option for a user.
3. Tags system to link posts with similar topics within one category.
4. Filtering option to divide and group posts and topics and ease the search option for a user.
5. Social media sharing option to stimulate and ease the content distribution.

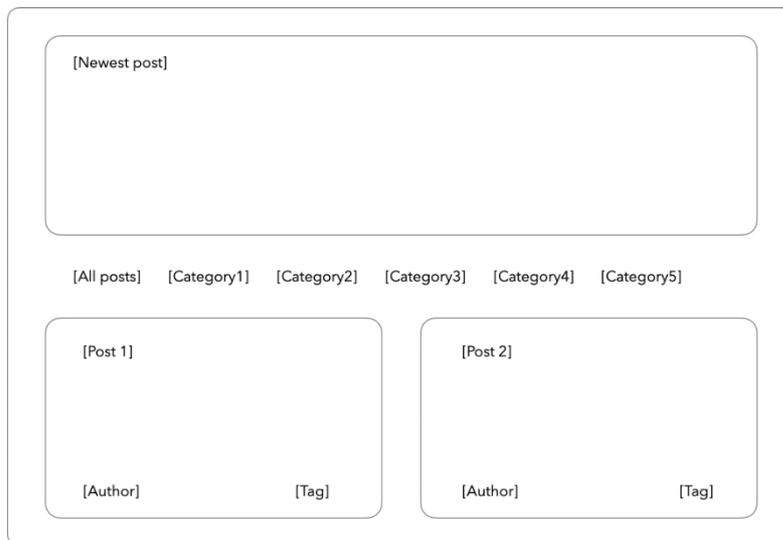
The provider should install and integrate a digital blog according to the visual identity of the project and detailed mock-ups granted by the Council of Europe. This blog should be inclusive and suitable for people with visual impairments.

The provider should also integrate the system within the AWS infrastructure the details of which will be communicated to the provider upon selection. This includes:

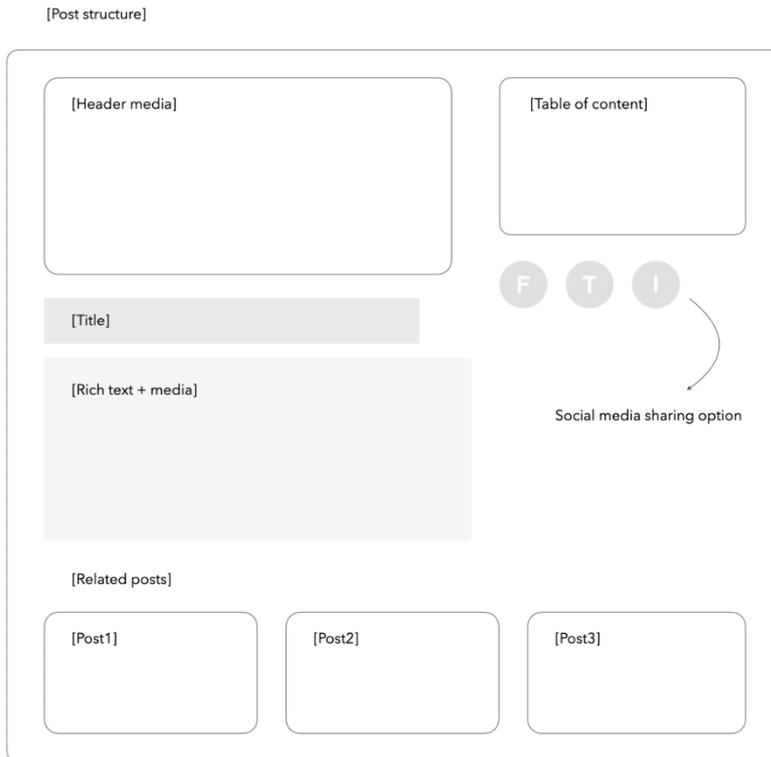
- providing the documentation of the needed infrastructure to the clients' system administrators so they can set up all that is needed for the future platform integration
- integrating the system on the provided infrastructure and providing basic instructions of its usage and maintenance

(prototype 14: digital blog)

[Digital blog]



(prototype 15: post structure)



Common part for all cases

Parameters

In the upper right corner of the main page a button with the user's avatar [Parameters] shall be placed. On pressing [Parameters], it shall be expanded to show options [Go to Dashboard], [My Settings], [Toggle Dark Mode] and [Log out the system].

On pressing [Go to Dashboard] the user is redirected to their dashboard (list of courses in progress and completed courses).

On pressing [My Settings] the user is redirected to the settings page where they can change their name to display and user picture.

On pressing [Toggle Dark Mode] the system changes between dark and light mode offering thus, an adaptive interface to the day and night light.

On pressing [Log out the System], the User shall exit the System.

Various categories of important requirements

1. The program shall be fully adapted for blind individuals and those with impaired vision; this envisages the following:

- The accessible program for the blind shall contain the components that will enable the users to use the screen reader programs with which they will have full access to the electronic training functions and receive the published information without hindrance. Every new information shall be separated by the headings. Menu elements, sub-menus and the forms shall be arranged in a manner so that they can be differentiated and perceived by the screen reader programs.
- To ensure the accessibility of the electronic training program for individuals with impaired vision, the following properties shall be considered: background – dark black colour; font colour - white, green, yellow; font size – indefinite.

It is desirable that the CEC online Platform is compatible with Wcag web content accessibility guideline 2.0 International standard.

2. System design – Council of Europe will grant the Provider with Platform visual identity and brand book for the CEC online Platform.

3. The System should have a capacity of safe operation with at least 50,000-70,000 users during 2 months after the launch of the System, and with at least 5000 users during the following 10 months.

4. The CEC online Platform should be developed in Ukrainian language.

5. The Provider shall create a guideline for using the system and familiarize the system recipients with it, as well as provide instructions in the area of the utilization and management of the system to be handed over with all the source codes of the online Platform to the Council of Europe and the Central Election Commission of Ukraine.

Ensuring the security of the web application

The application shall consider the following issues:

1. Latest stable versions of the libraries shall be used in the application.
2. The information entered by the user shall be filtered and sanitized.
3. None of the Error messages shall contain information about the system or related to it.
4. Access handling:

4.1. *Authorization.* OAuth 2.0 protocol should be used for authentication. It should use password and external authentication (Facebook) grant types that should expire in 24h and then it should use refresh token to refresh the authorization token.

4.2. *Access control*

4.2.1. All users (including system users) shall have access only to the information that is necessary for their operations.

4.2.2. Upon the request of accessing each resource, its validity shall be verified.

4.3. *Injections. SQL injection*

4.4. *Database level restrictions.* Different users shall be created based on various types of users. Moreover, libraries used in the application shall ensure preventing the following threats:

4.4.1. SMTP injection

4.4.2. Cross-site scripting (XSS)

4.5. Upon completion of the project, application's "source" code shall be delivered to the recipient of the System.

5. Web application shall use the following technologies:

5.1. *Java programming language with Spring framework*

5.2. *Javascript, HTML, CSS, Vue.js or React framework*

5.3. *Databases: MySQL / DynamoDb / Elasticsearch*

Analytics and monitoring

The system should provide the following information per each user, per each course and per lecture:

1. the number of enrolled users
2. the number of users who received course certificate
3. the time needed for the course completion

It should be possible to drill down the information by time (weeks, days, hour).

The provider should also install Google Analytics to the project subpages and digital blog.