Appendix I to the Act of Engagement
Technical description of the services to be provided

Website support, frontend and backend design services related to the maintenance and development of the online awareness-raising platform "CEC: prosvita"

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.

In 2020, upon the request of the Central Election Commission of Ukraine (CEC) – Project main national partner, the Council of Europe supported the development and launch of the online Platform "CEC: prosvita" available upon the link https://www.cvkpro.com/en.

The "CEC: prosvita" online Platform (https://www.cvkpro.com/en) 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.

As of now, the "CEC: prosvita" online Platform contains 2 online training courses on elections for voters placed thereon ("Basic level course "Elections in simple terms" and Advanced level course "Local elections for voters"), as well as awareness-raising and educational materials about various aspects of the electoral process. Considering the CEC request to further support the maintenance and development of the "CEC: prosvita" online platform, the Project is now planning to select providers for specifically web development services for the "CEC: prosvita" online platform, particularly, with regard to website support, frontend and backend design services.

Partnership: The "CEC: prosvita" online Platform was 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".

Overall objective of the "CEC: prosvita" 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 further 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: prosvita" 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: prosvita" online Platform with further digital products placed thereon/introduced therein is in Ukrainian and available for any Ukrainian. For these purposes, the "CEC: prosvita" online Platform is linked to the CEC official website.

Methodology:

The distant platform enables users to access the "CEC: prosvita" online Platform without any time and space limitations.

The navigation between the different e-learning formats within one content block is 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 also includes 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 has 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 happens in the WYSIWYG (what you see is what you get) way, meaning that the course administrator sees everything as it is displayed to the user, basically editing all the course information in online mode.

Platform Inclusiveness: The online platform is fully adapted for people with visual and hearing (production of subtitles on video products) impairments.

Statistical Data Processing Capacity: The software platform of the course has technical capacity for data processing (report) that enables 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 is possible to drill down the information by time (weeks, days, hour). Processed statistical data (protection of personal data is ensured) is used for program evaluation purposes.

Technical assignment for the following services :

- LOT 1 (A). Frontend design services related to the development and maintenance of the "CEC: prosvita" online Platform;
- LOT 2 (B). Backend design services related to the development and maintenance of the "CEC: prosvita" online Platform;
- LOT 3 (C). Website support of the "CEC: prosvita" online Platform (daily backups, system upgrades, security protection

is provided below.

Technical Assignment for the “CEC: prosvita” online Platform

A. Graphical design and front-end development of the client side of the “CEC: prosvita” online Platform

1. Context

The “CEC: prosvita” online Platform should give users the possibility to interact with educational and information resources. User interaction should be done via browser based web application(portal). To deliver this functionality to end users the “CEC: prosvita” online Platform needs to be visually designed and implement the front-end part of the portal in a way to support devices with most common screen form factors and resolution such as cell phones, tablets, personal computers/laptops.

2. After release support

Due to the nature of the requested tasks, the development process may involve active communication with other developers/teams that may be also involved in implementation back-end/design/front-end functionality of the “CEC: prosvita” online Platform. Because of possibilities that some parts of the “CEC: prosvita” online Platform may be strongly dependent from each other but implemented by different developers/teams during different timelines, the Council of Europe can request additional work on already submitted tasks. The Council of Europe reserves the right to request additionally up to 10% of the total initial task execution time to resolve critical issues that have direct impact on functionally/performance/stability/security or other developer/teamwork. Requests for additional work after tasks that have already been accepted may be done during 3 months.

Key point and idea are to deliver a well performed solution that is fulfilling user needs, in order to achieve this the Council of Europe encourages all developers/teams to suggest the best approach for all tasks.

3. Technical description

Front-end: The “CEC: prosvita” online Platform’s frontend application is a SPA (single page application) written using the HTML and Nuxt.js web application framework. It is based on Vue.js, Node.js, Webpack and Babel.js. The application’s state is managed using Vuex – a state management library for Vue.js that serves as a centralized store for all the components in an application, with rules ensuring that the state can only be mutated in a predictable fashion.

The Frontend application communicates with the backend using its REST API. The data are transformed on the data layer being lately consumed by the Vue application using the interface between Vue and Vuex.

The view layer is data agnostic and leverages the data reactiveness (the underlying concept of Vue application) to update data on the web page.

All delivered code should come with meaningful comments and comply with common standards for front-end development.

Design: All visual and graphical resources should be part of task submissions and should come in proper formats and include source files and compressed output such as PNG, JPEG, PDF or SVG in a case of graphical resources.

4. Documentation

All delivered code should be stored in version control system like GitHub and come with fulfilled documentation to cover next cases:

- description what has been done and what it may effect;
- deploying process;
- user guidance;
- API documentation to describe what kind of request/response expected;
- code comments to explain logic;
- list of external resources that been used;

5. Third party resources usage

Task execution may require usage of third party resources, code, fonts etc. All external resources must be free to use or distribute under any kind of "free licence" such as Apache, MIT etc. Also any kind of external code that has been used and under free licence should be listed in documentation with reference and be maintained for at least for 1 year.

6. Key points

- maintaining and improving the "CEC: prosvita" online Platform;
- develop new features for the "CEC: prosvita" online Platform;
- collaborating with backend developers and web designers to improve usability;
- getting feedback and build solutions for users;
- writing functional requirement documents and guides;
- creating quality mockups and prototypes;
- helping backend developers with coding and troubleshooting;
- ensuring high quality graphic standards and brand consistency;

7. Development process

Development process should include code submission to GitHub with code review and ensuring that no unused/redundant/fragile code is submitted. The Council of Europe can establish periodic or on-demand demos to track task execution process.

B. Development and deployment of the server side of the "CEC: prosvita" online Platform

1. Context

The "CEC: prosvita" online Platform provides two kinds of access for users. Unauthorised access to get not restricted and public information. On the other hand authorised access only for registered users and user specific information. All data requests transferred from the front-end part of the "CEC: prosvita" online Platform to the back-end part which hosted on AWS. To provide a properly functioning back-end requires additional development and implementation of new features. Some features will include data design creation and implementations to support content management systems.

2. After release support

Due to the nature of the requested tasks, the development process may involve active communication with other developers/teams that may be also involved in implementation back-end/front-end functionality of the "CEC: prosvita" online Platform. Because of possibilities that some parts of the "CEC: prosvita" online Platform may be strongly dependent from each other but implemented by different developers/teams during different timelines the Council of Europe can request additional work on already submitted tasks. The Council of Europe reserves the right to request additionally up to 10% of the total initial task execution time to resolve critical issues that have direct impact on functionally/performance/stability/security or other developer/teamwork. Requests for additional work after tasks that have already been accepted may be done during 3 months.

3. Technical description

The the "CEC: prosvita" online Platform's backend is a monolithic application written in Java 11 and using the Spring framework that communicates with data storages. It is a RESTful application that exposes endpoints to access services and data. Also, back-end hosted on AWS.

In order to store data, the application makes use of two databases: a MySQL relational database to store all the business-critical data, and a NoSQL database (DynamoDB) to store authentication information such as access and refresh token information.

All delivered code should come with meaningful comments and comply with common standards for back-end development and be fully backwards compatible. Submitted code should have unit/integration test coverage be scalable and tolerant to user data.

4. Documentation

All delivered code should be stored in version control system like GitHub and come with fulfilled documentation to cover next cases:

- description how to revert to previous version and redeploy if necessary;
- description what has been done and what it may effect;
- deploying process;
- user guidance;
- API documentation to describe what kind of request/response expected;
- list of external libraries and versions that have been used;

All the documentation should be provided in English. Ukrainian versions may be requested too, upon the needs related to the Project implementation and the requests of the Project national partner – Central Election Commission of Ukraine.

5. Third party resources usage

Task execution may require usage of third party resources, code etc. All external resources must be free to use or distribute under any kind of "free licence" such as Apache, MIT etc. Also, any kind of external code that has been used and under free licence should be listed in documentation with reference and be maintained for at least for 1 year.

6. Key points

- database creation, integration, and management;
- working with web server technologies, cloud computing integration;
- content management system development, deployment, and maintenance;
- API integration;
- maintaining and improving the Platform;
- ensuring security and user data protection;
- reporting – generating analytics and statistics;
- backup and restore technologies for a website's files and databases;
- collaborating with the frontend developers and other team members to establish objectives and design more functional, cohesive codes to enhance the user experience;
- developing ideas for new solutions;
- writing detailed documentation;
- design system with decent level of performance and scalability;

C. Website support of the "CEC: prosvita" online Platform (daily backups, system upgrades, security protection, infrastructure maintenance)

1. Context

The "CEC: prosvita" online Platform is a cloud-based web application that contains back-end and front-end parts. All those parts are hosted on AWS and utilise few databases. To provide stable operation and data safety of the platform some regular maintenance work should be performed.

2. After release support

Due to the nature of the requested tasks, the development process may involve active communication with other developers/teams that may be also involved in implementation back-end/front-end functionality of the "CEC: prosvita" online Platform. Because of possibilities that some parts of the "CEC: prosvita" online Platform may be strongly dependent from each other but implemented by different developers/teams during different timelines the Council of Europe can request additional work on already submitted tasks if critical issues were found.

3. Technical description

The "CEC: prosvita" online Platform requires setting up automatic maintenance jobs such as database backups for MySQL and DynamoDB. Also setting up auto scaling and system failure protection and other tools that provide system status monitoring and protection.

List of maintenance work:

- monitoring for the "CEC: prosvita" online Platform's needs to set up event logging system to monitor errors and other system events;
- Google analytics;
- regular DB damps;
- framework version updates;
- access management for developers and managers;
- AWS infrastructure maintenance;
- errors and issues fixes;
- security issues monitoring;
- reporting – generating analytics and statistics;
- collaborating with the other developers and other teams to establish objectives and design more functional, cohesive solutions to enhance the user experience;
- recording data and reporting it to proper parties, such as clients or leadership;
- managing version control systems like GitHub;
- system monitoring and creating list of issues that currently or potentially may cause a problem;

4. Documentation

- create report after every maintenance task with list of all changes and fixes;
- reporting potential problems;

5. **Third party resources usage**

All resources used for task execution must be free to use or with valid licence.

Common part for all cases

Data/privacy/security/intellectual property regulation

Development process may include situations where it is required to use user sensitive data. All interaction with user private data must be complied with the Council of Europe Modernized Convention for the Protection of Individuals with regard to Automatic Processing of Personal Data (Convention 108+) https://www.coe.int/en/web/data-protection/convention108/modernised and GDPR (General Data Protection Regulation) https://gdpr-info.eu/. Any usage of user sensitive data must be agreed with the Council of Europe and deleted after task competition. The list of user sensitive data includes but is not limited to passwords, addresses, emails, phone numbers etc.

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. The granted access should expire in 24h and then it should use a 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. *Prevent 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.

Performance

All deployments or new solutions should be without downtime or in a case if a task cannot be completed without downtime this should be confirmed with the Council of Europe.

Any task execution should not downgrade response time for existing APIs or break current functionality.

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 the "CEC: prosvita" online Platform visual identity and brand book for the "CEC: prosvita" 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: prosvita" 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.

User 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.