**Corporate Services Division / Enterprise Solutions Section**

**Written Assessment for National UNV**

**Application Development and Support**

**Instructions**

Please take look at the attached Excel sheet listing some projects undertaken by UNEP and answer the following questions (Section I).

Time limit: **24 hours**.

**Do not plagiarize.**

Kindly return the completed assessment by **12.30 hrs, Saturday, 15 October 2022.**

**Start of Assessment**

**Section I**

1. Normalize the data in the Excel and show relationships between the derived tables, if any. *Use of primary keys, foreign keys etc. is strongly encouraged.*
2. Create a web interface (using HTML forms) that can be used to manage these projects i.e. add, edit and delete projects (CRUD functionalities). *For the sake of time, only the project’s table needs to have an interface for CRUD operations. The other derived tables can be managed from the backend.*
3. Use of front-end libraries and frameworks like Bootstrap, FontAwesome, VueJS, AngularJS, ReactJS, etc. is strongly encouraged but not necessary.
4. The back-end should be done using PHP, Python, NodeJS or any other web programming language that runs on a Linux environment. Use of frameworks like PhalconPHP, Laravel, Django, ExpressJS is strongly encouraged, but not necessary.
5. The database can be MySQL, MariaDB, SQLite, MongoDB or any other Database system that runs on a Linux environment.
6. Create a page that lists these projects in a tabular format.
7. The table should display 10 rows per page by default, but with the option of allowing the user to choose the number of rows to display e.g. 5, 20, 50, all etc.
8. There should be a pager at the bottom of the page showing the current page number e.g. page 3 of 5 or something similar.
9. For each record, there should be a column for actions (view, edit, delete) which when clicked allows the user to view the record in detail, edit the record or delete the record.
10. Create an API that exposes the above projects as JSON. The API endpoints can work as explained below:
11. Navigating to a URL like *api/projects/all* will return all projects as JSON
12. Navigating to a URL like *api/projects/country/kenya* will return all projects from country Kenya as JSON
13. Navigating to a URL like *api/projects/status/completed* will return all projects with status completed as JSON
14. **Important**: Share your source code, database dumps and **instructions on how to install your application** via **GitHub, GitLab or Bitbucket**

**Section II**

1. What should be the features of a good collaboration and business process automation software according to you?

A good collaboration and business process automation software should be;

1. User-friendly in terms of easy to navigate by a project member or user of the tool.
2. It should be flexible in order to make crucial changes that may occur later in a project.
3. It should have a clear workflow that allows the integration, recording and utility of data and information.
4. It should be compatible with other integration tools that improve collaboration within project teams and other users.
5. What is the aim, benefits and limitations of the Confluence and JIRA tools?

Confluence is a tool that is used by a group of people for collaboration and knowledge management and transfer. The aim for the tool is to efficiently enhance teams working on various projects by assigning roles to users of a project team, gathering project requirements and other tasks such as project timelines while the JIRA tool is used for project monitoring and management.

The benefit of integrating these two tools is to help a project team to enhance collaboration, sharing of data and information and improve on project processes making it more deliverable at the stated period.

The limitation of the two tools are quite clear as Confluence tool is more rigid due to its procedural methodologies of storing information in a page while JIRA tool is not user-friendly hence navigation to various functions is quite hard.

1. Suggest practical ways of addressing performance issues with Confluence, for example, in case the system page load is too slow.
2. The RAM capacity can be increased to mitigate the issue of slow page loading.
3. Streamline other resources that have been integrated with the Confluence Tool hence improving its page load as other unnecessary processes are not initiated.
4. Ensure that the Confluence cache size is tuned in order to improve the page load.
5. Eradicate bottleneck scenarios on the database by checking its latency and optimize on its performance to improve the tool’s loading page speed.
6. Please be as creative as possible.

-End-