EMIS Software Development Options

Option 1: Proprietary Software Solution. MoE purchases an EMIS solution from a software development company. The company develops or configures the system for the national education system.

Pros: Customized to meet current functional requirements of MoE; if off-the-shelf then, may be immediately available for implementation

Cons: Licensing fees may be expensive and recurrent; requests for future changes and enhancements to the system must be done by the software company and may be expensive; if bespoke development is required, then the system may take several months to be ready for implementation; lack of ownership of the system by the MoE; dependence of MoE on a private software company for EMIS

Option 2: In-house Software Solution. MoE hires software programmers as part of its team to develop and deploy the national EMIS.

Pros: Customized to meet current functional requirements of MoE; requests for future changes and enhancements to the system can be done independently by the MoE software development staff; ownership of the system by the MoE

Cons: Challenges to develop robust software solutions in-house with limited resources for full software development cycle (functional requirements, system architecture, system specifications, user-interface design, mockups, prototyping, quality assurance testing, fine-tuning, user acceptance testing, deployment, service desk technical support, system maintenance, technical documentation, user documentation, training and capacity building); high-risk of dependence on a small team of programmers that may eventually leave the MoE; narrow scope from building on EMIS lessons learned in other countries;

Option 3: Open Source Software Solution. MoE licenses open source software which is an off-the-shelf generic EMIS that can be configured and deployed as the national EMIS.

Pros: No licensing fees; customized to meet current functional requirements of MoE; requests for built-in customizable features of the EMIS can be done by MOE administrative staff who can change the configuration of the software without changing the underlying software code; full ownership of the system by the MoE independent of external technical support; continuous real-time improvement by the community of users; new features can be added to the system by the multinational community of users and shared with all other users of the system who have agreed to collaborate with ideas, resources and software enhancements; this off-the-shelf option can be configured and deployed rapidly

Cons: Long-term commitment by MoE required for independent administration of the system without external assistance; open source software requires resources for capacity building, configuration, implementation, and technical support costs.

OpenEMIS Initiative

- OpenEMIS software is made freely available to members under an open source *Common Development and Distribution License* (Version 1, January 2005);

- OpenEMIS is designed as a high-aligned, loosely coupled system of interoperable products for data capture, management, and analysis, which are designed to either work independently or as part of the entire OpenEMIS architecture ecosystem allowing the MoE to choose the set of products that meet their needs.

OpenEMIS Principles for Innovative Software Development

1. Design with the User

- Develop context appropriate solutions informed by user needs.
- Include all user groups in planning, development, implementation and assessment.
- Develop projects in an incremental and iterative manner.
- Design solutions that learn from and enhance existing workflows and plan for organizational adaptation.
- Ensure solutions are sensitive to, and useful for, the most marginalized populations: women, children, those with disabilities, and those affected by conflict and disaster.

2. Understand the Existing Ecosystem

- Participate in networks and communities of like-minded practitioners.
- Align to existing technological, legal, and regulatory policies.

3. Design for Scale

- Design for scale from the start, and assess and mitigate dependencies that might limit ability to scale.
- Employ a “systems” approach to design, considering implications of design beyond an immediate project.
- Be replicable and customizable in other countries and contexts.
- Demonstrate impact before scaling a solution.
- Analyze all technology choices through the lens of national and regional scale.
- Factor in partnerships from the beginning and start early negotiations.

4. Build for Sustainability

- Plan for sustainability from the start, including planning for long-term financial health i.e., assessing total cost of ownership.
- Utilize and invest in local communities and developers by default and help catalyze their growth.
- Engage with local governments to ensure integration into national strategy and identify high-level government advocates.

5. Be Data Driven

- Design projects so that impact can be measured at discrete milestones with a focus on outcomes rather than outputs.
- Evaluate innovative solutions and areas where there are gaps in data and evidence.
- Use real-time information to monitor and inform management decisions at all levels.
- When possible, leverage data as a by-product of user actions and transactions for assessments.

6. Use Open Standards, Open Data, Open Source, and Open Innovation

- Adopt and expand existing open standards.
- Open data and functionalities and expose them in documented APIs (Application Programming Interfaces) where use by a larger community is possible.
- Invest in software as a public good.
- Develop software to be open source by default with the code made available in public repositories and supported through developer communities.

7. Reuse and Improve

- Use, modify and extend existing tools, platforms, and frameworks when possible.
- Develop in modular ways favoring approaches that are interoperable over those that are monolithic by design.

8. Do no harm

- Assess and mitigate risks to the security of users and their data.
- Consider the context and needs for privacy of personally identifiable information when designing solutions and mitigate accordingly.
- Ensure equity and fairness in co-creation, and protect the best interests of the end end-users.

9. Be Collaborative

- Engage diverse expertise across disciplines and industries at all stages.
- Work across sector silos to create coordinated and more holistic approaches.
- Document work, results, processes and best practices and share them widely.