ІНФОРМАЦІЙНА СИСТЕМА ЕКСПЕРТИЗ РАДІАЦІЙНОЇ БЕЗПЕКИ

О. М. Ткаченко, кандидат технічних наук, доцент

E-mail: otkachenko@nubip.edu.ua

В. М. Куценко, студент магістратури

Національний університет біоресурсів і природокористування України

E-mail: vadim.nik.kutsenko@gmail.com

Анотація. *Радіаційна безпека є однією із важливих складових екологічної безпеки загалом. Радіаційний контроль є необхідним не лише в контексті виробництва і використання ядерної зброї, а й для країн, які мають розвинену інфраструктуру атомної енергетики або постраждали від радіаційних викидів (Україна, Білорусія, Японія та інші). Процес управління в цій сфері є досить наукоємним і, зрозуміло, вимагає ретельного обліку всіх дозвільних документів, висновків та звітності.*

В Україні функціонує Державний науково-технічний центр з ядерної та радіаційної безпеки (ДНТЦ ЯРБ). Він здійснює науково-дослідну діяльність, готує нормативні документи, експертні висновки у зазначеній сфері. На підприємстві функціонувала застаріла інформаційна система, призначена для обліку, контролю термінів і стану виконання експертиз і договорів, підготовки звітності.

Метою дослідження є виявлення недоліків існуючої системи обліку експертиз ДНТЦ ЯРБ, проектування та реалізація програмного забезпечення, яке вирішить проблеми в наявній системі обліку експертиз у сфері ядерної та радіаційної безпеки.

У цій статті пропонується опис технологій модернізації цієї системи, зокрема, запропоновано оновлену структуру, розширений функціонал, програмну реалізацію окремих модулів.

Програмне, апаратне забезпечення та окремі програмні рішення в рамках завдань відповідають стандартам життєвого циклу розробки програмного забезпечення, організації баз даних, верифікації і документування. Інструментальною платформою розробки є .NET і мова C#.

Ключові слова: *радіаційна, безпека, система, експертиза, облік*

Актуальність. *Державний науково-технічний центр з ядерної та радіаційної безпеки (ДНТЦ ЯРБ) є базовою експертною організацією Держатомрегулювання і за його завданням здійснює розгляд наданих Замовником матеріалів, перевірку відповідності технічних рішень вимогам чинних нормативно-правових актів України, норм правил і стандартів з ядерної та радіаційної безпеки, аналіз повноти й*

достовірності вихідних даних і припущень, коректності проведених розрахунків, отриманих результатів та висновків з метою оцінки безпеки у сфері використання ядерної енергії.

Інформаційно-довідкове забезпечення на підприємстві ґрунтується на базах даних, які функціонують у єдиній інформаційній системі. Розрахунково-аналітичне забезпечення експертної діяльності об'єднує близько 40 програмних засобів. Обсяг експертної діяльності щорічно становить близько 70 % обсягу всієї науково-технічної діяльності ДНТЦ ЯРБ [1].

На підприємстві функціонувала застаріла автоматизована інформаційна система "Експертизи", призначення якої – облік, контроль термінів і стану виконання експертиз і договорів, отримання різноманітних звітів і довідок на основі існуючих даних для проведення своєчасного аналізу. Системою активно користуються близько 100 користувачів (експертів, начальників структурних підрозділів, операторів ведення експертиз та договорів).

Наявна система обліку експертиз не задовольняє запити підприємства з точки зору функціональності, надійності, швидкості роботи, придатністю роботи на різних платформах, безпеки даних. Конструкція запитів веде до нераціонального використання обчислювальних ресурсів, що призводить до уповільнення роботи інших програмних продуктів підприємства, розміщених на тому ж сервері. Прикладом є уповільнення виконання робочих циклів системи електронного документообігу на базі SharePoint, що неодноразово призводило до системних збоїв. Основні проблеми існуючої системи пов'язані з:

- стабільністю роботи;
- швидкістю виконання робочих процесів;
- оптимальністю використання обчислювальних ресурсів;
- адаптованістю для різних пристроїв;
- актуальністю застосованих технологій розробки.

Аналіз досліджень і публікацій. Багато публікацій наукової спільноти і розробників присвячено створенню спеціалізованих систем моніторингу стану

радіаційної безпеки в контексті загроз роботи атомних станцій та інших ядерних об'єктів. Так, у [2] запропоновано інформаційну систему радіаційного моніторингу IRMIS, яку розробило Міжнародне агентство з атомної енергії веб-орієнтовану для обміну і візуалізації даних радіаційного моніторингу. Дані відображають області потенційного поширення радіації, що покликане допомогти країнам вжити відповідні дії у надзвичайних ситуаціях. Ряд статей, наприклад [3], присвячені проектуванню експертних систем оцінки ризиків для довкілля від діяльності атомних електростанцій. У [4] описано архітектуру та імплементацію геопросторової інформаційно-управляючої системи підтримки проведення заходів моніторингу та попередження ризиків радіаційної враження для спеціалістів, що працюють у дослідницьких центрах і службах радіаційного захисту. Стандарти обліку руху документів у цих системах відповідає національним нормативним документам. Стержнем багатьох систем радіаційного моніторингу є геопросторовий продукт [5], який може бути складовою комплексної системи, наприклад, у сільському господарстві [6].

Метою дослідження є виявлення недоліків існуючої системи обліку експертиз ДНТЦ ЯРБ, проектування та реалізація програмного забезпечення, яке вирішить проблеми в наявній системі обліку експертиз у сфері ядерної та радіаційної безпеки.

Матеріали та методи дослідження. Програмне, апаратне забезпечення та окремі програмні рішення в рамках завдань відповідають стандартам життєвого циклу розробки програмного забезпечення, організації баз даних, верифікації і документування. Інструментальною платформою розробки є .NET і мова C#.

Результати досліджень та їх обговорення. У процесі створення системи насамперед було вирішено проблему авторизації з використанням Active Directory (AD).

Ідентифікація – процес визначення, з якою особою має справу система, *автентифікація* – процес підтвердження, що ця особа є тим, за кого себе видає, *авторизація* – процес прийняття рішення, що саме автентифікованій персоні

дозволяється робити. Існуюча система, і, відповідно, автентифікація була побудована на базі MS Access 2010 (рис. 1).

Рис. 1. Реалізація автентифікації засобами Access 2010

На підприємстві існує близько 6 подібних систем, тому такий підхід є неприйнятним. Хоча в таблиці присутнє поле "tPsw", паролі там не зберігаються, вони синхронізуються з AD. Дублювання інформації та роздроблення компонентів, які знаходяться в прямій залежності один від одного, є недоліком. З одного боку, це записи в БД, які необхідно щоразу вручну редагувати для управління правами користувачів, з іншого – можна використовувати більше можливостей AD. Чи не найпершою задачею при створенні оновленої системи було проектування та реалізація централізованої точки зберігання інформації про користувачів, що уніфікувало доступ до цих даних з інших програмних продуктів. Вибір серверу AD, який містив всю основну інформацію про працівників, став одним з оптимальних рішень.

Другою задачею була реалізація функціоналу в новій системі з використанням AD і технології ідентифікації та авторизації. Інструментом розробки нової системи було обрано .NET ASP.NET MVC 5. На момент реалізації обрана бібліотека Galactic.LDAP, яка базується на .NET Framework. Хорошою альтернативою цьому є System.DirectoryServices.AccountManagement від Microsoft.

Отже, з використанням AD і Galactic.LDAP ролі і групи користувачів зберігаються у вигляді, поданому на рис. 2.

Рис 2. Групи AD, в яких знаходиться користувач

Для реалізації ASP.NET MVC 5 необхідно було перевизначити стандартний провайдер ролей RoleProvider, а саме: методи GetRolesForUser та IsUserInRole. Перший отримує всі групи користувача в AD, а другий на основі отриманих груп визначає, чи є в деякого користувача певна роль (рис. 3).

```
public override string[] GetRolesForUser(string username)
{
    var allRoles = new List<string>();
    var root = new DirectoryEntry(path: WebConfigurationManager.ConnectionStrings[ConnectionStringName].
        ConnectionString, ConnectionUsername, ConnectionPassword);
    var searcher = new DirectorySearcher(root, string.Format(CultureInfo.InvariantCulture,
        "&(objectClass=user)({0}={1})", AttributeMapUsername, username));
    searcher.PropertiesToLoad.Add(value: "memberOf");
    SearchResult result = searcher.FindOne();
    if (result != null && !string.IsNullOrEmpty(result.Path))
    {
        DirectoryEntry user = result.GetDirectoryEntry();
        PropertyValueCollection groups = user.Properties["memberOf"];
        foreach (string path in groups)
        {
            string[] parts = path.Split(params separator: ',');
            if (parts.Length > 0)
            {
                foreach (string part in parts)
                {
                    string[] p = part.Split(params separator: '=');
                    if (p[0].Equals(value: "cn", StringComparison.OrdinalIgnoreCase))
                    {
                        allRoles.Add(p[1]);
                    }
                }
            }
        }
    }
}

public override bool IsUserInRole(string username, string roleName)
{
    bool isRole = false;
    var roles = GetRolesForUser(username);
    foreach (var item in roles)
    {
        if (item == roleName)
        {
            isRole = true;
        }
    }
    return isRole;
}
```

Рис. 3. Перевизначений RoleProvider

Подальше управління доступом відбувається на рівні контролерів за допомогою використання атрибутів авторизації. Перед контролером чи методом треба перерахувати групи, які мають доступ до даного функціоналу (рис. 4).

```
[HttpPost]  
[Authorize(Roles = "DBExpert_Administrator, DBExpert_Write1")]  
public ActionResult Delete(int id)
```

Рис 4. Використання функціоналу в контролері

Ще однією проблемою попередньої версії системи була організація БД. Фактично вона складалася з однієї результуючої та 13 довідкових таблиць, без урахування тих, які використовуються для управління користувачами та доступом. Цілісність даних забезпечена не була.

Як один з варіантів, проблему зв'язності можна виправити за допомогою одного з популярних ORM фреймворків. Entity Framework Fluent API використовується для налаштування доменних класів та перевизначення умов їх використання. Можливості Fluent API також можна використати для конфігурації схеми даних БД під час виконання програми. Використовуючи Entity Framework, класи моделей зіставляються з таблицями за допомогою ряду правил. Іноді необхідно змінити і перевизначити логіку цих правил. Для цього використовуються Fluent API та анотації даних. Fluent API є набором методів, що визначають зіставлення між класами і їх властивостями та таблицями з їх стовпцями. Як правило, функціонал Fluent API використовується при перевизначенні методу OnModelCreating. На рис. 5. Подано реалізацію зв'язку один до нуля або до одному (One-to-Zero-or-One).

Метод HasRequired () вказує, що для сутності Phone обов'язково повинно бути вказано навігаційну властивість Company. А метод WithOptional (), навпаки, встановлює необов'язковий зв'язок між об'єктом попереднього виразу і його властивістю BestSeller. Рис. 6 демонструє реалізацію зв'язків 1–1 та 1–N.

```
protected override void OnModelCreating(DbModelBuilder modelBuilder)
{
    modelBuilder.Entity<Phone>()
        .HasRequired(c => c.Company)
        .WithOptional(c => c.BestSeller);

    base.OnModelCreating(modelBuilder);
}
```

Рис. 5. Відношення One-to-Zero-or-One

```
modelBuilder.Entity<Phone>()
    .HasRequired(c => c.Company)
    .WithRequiredPrincipal(c => c.BestSeller);
```

```
modelBuilder.Entity<Phone>()
    .HasMany(p => p.Companies)
    .WithMany(c => c.Phones);
```

Рис. 6. Реалізація зв'язків 1–1 та 1–N

Метод `WithRequiredPrincipal()` налаштовує обов'язковий зв'язок і встановлює одну з сутностей як основну. Так, в даному випадку основна як сутність встановлюється модель `Phone`: `WithRequiredPrincipal (c => c.BestSeller)`. А таблиця, на яку відображається модель `Company`, буде містити зовнішній ключ до таблиці `Phones`. `HasMany ()` встановлює множинний зв'язок між об'єктом `Phone` і об'єктами `Company`. А метод `WithMany ()` додає обернений множинний зв'язок між об'єктом `Company` і об'єктами `Phone`.

Для налаштування бази засобами `Fluent API` за допомогою підходу `DatabaseFirst` було згенеровано множину класів, що відповідає сутностям бази даних (рис. 7).

Властивості класів, які відповідають таблицям БД, вказані як `virtual` – це налаштування дозволяє застосувати "ліниве" завантаження даних, коли

завантаження відбувається при першому зверненні до об'єкта, а якщо пов'язані дані не потрібні, вони не завантажуються (рис. 8).

```
public class ExpertContext : System.Data.Entity.DbContext
{
    public virtual DbSet<tbExpertizy> tbExpertizy { get; set; }
    public virtual DbSet<tbIndexVD> tbIndexVD { get; set; }
    public virtual DbSet<tbIspolnitel> tbIspolnitel { get; set; }
    public virtual DbSet<tbNapravlenieER> tbNapravlenieER { get; set; }
    public virtual DbSet<tbOrganizatsiya> tbOrganizatsiya { get; set; }
    public virtual DbSet<tbOtdelenie> tbOtdelenie { get; set; }
    public virtual DbSet<tbRamDogovor> tbRamDogovor { get; set; }
    public virtual DbSet<tbSoispolniteli> tbSoispolniteli { get; set; }
    public virtual DbSet<tbStatusD> tbStatusD { get; set; }
    public virtual DbSet<tbStatusE> tbStatusE { get; set; }
    public virtual DbSet<tbTipDogovora> tbTipDogovora { get; set; }
    public virtual DbSet<tbVidER> tbVidER { get; set; }
    public virtual DbSet<tbVidRS> tbVidRS { get; set; }
}
```

Рис. 7. Визначення контексту бази даних

```
[Table("tbExpertizy")]
public class tbExpertizy
{
    public int ID { get; set; }
    public int? N1 { get; set; }
    public int? N2 { get; set; }
    public int? N3 { get; set; }
    public int? nZayavitel { get; set; }
    public int? nZakazchik { get; set; }
    public int? nOtdelenie { get; set; }
    public int? nIspolnitel { get; set; }
    public int? nVidRS { get; set; }
    public int? nNapravlenieER { get; set; }
    public int? nTipDogovora { get; set; }
    public int? nIndexVD { get; set; }
    public int? nVidER { get; set; }
    public int? nStatusE { get; set; }
    public int? nStatusD { get; set; }

    [StringLength(50)]
    public string tLetterZak { get; set; }

    public DateTime? dLetterZak { get; set; }

    [StringLength(50)]
    public string tLetterZayav { get; set; }

    public DateTime? dLetterZayav { get; set; }
}
```

Рис. 8. Частина згенерованого класу

Далі перевизначається метод `OnModelCreating`, що дозволяє сконфігурувати схему даних під час виконання веб-застосунку. У даному випадку налаштовуються зв'язки N–1 від довідкових таблиць до результуючої таблиці (рис. 9).

```
protected override void OnModelCreating(DbModelBuilder modelBuilder)
{
    modelBuilder.Entity<tbOrganizatsiya>()
        .HasMany(navigationPropertyExpression: e => e.tbExpertizys)
        .WithOptional(navigationPropertyExpression: e => e.tbOrganizatsiya)
        .HasForeignKey(e => e.nZayavitel);

    modelBuilder.Entity<tbOrganizatsiya>()
        .HasMany(navigationPropertyExpression: e => e.tbExpertizys2)
        .WithOptional(navigationPropertyExpression: e => e.tbOrganizatsiyaZak)
        .HasForeignKey(e => e.nZakazchik);

    modelBuilder.Entity<tbOtdelenie>()
        .HasMany(navigationPropertyExpression: e => e.tbExpertizys)
        .WithOptional(navigationPropertyExpression: e => e.tbOtdelenie)
        .HasForeignKey(e => e.nOtdelenie);

    modelBuilder.Entity<tbIspolnitel>()
        .HasMany(navigationPropertyExpression: e => e.tbExpertizys)
        .WithOptional(navigationPropertyExpression: e => e.tbIspolnitel)
        .HasForeignKey(e => e.nIspolnitel);

    modelBuilder.Entity<tbOtdelenie>()
        .HasMany(navigationPropertyExpression: e => e.tbIspolnitels)
        .WithOptional(navigationPropertyExpression: e => e.tbOtdelenie)
        .HasForeignKey(e => e.nOtdelenie);

    modelBuilder.Entity<tbNapravlenieER>()
        .HasMany(navigationPropertyExpression: e => e.tbExpertizys)
        .WithOptional(navigationPropertyExpression: e => e.tbNapravlenieER)
        .HasForeignKey(e => e.nNapravlenieER);
}
```

Рис. 9. Конфігурація зв'язків схеми

Висновки і перспективи. Запропонований підхід розробки оновленої версії системи обліку експертиз Державного науково-технічного центру з ядерної та радіаційної безпеки усуває ряд проблем, які були властиві попередній версії системи, зокрема, погана структурованість і забезпеченість цілісності даних, застарілість платформи, слабкий рівень захисту, ефективності взаємодії з іншими

програмними продуктами організації та навантаження на апаратне забезпечення і мережу. Серед подальших перспектив розбудови системи відзначимо поглиблення її інтеграції з іншим програмним забезпеченням, зокрема, модифікація інших програмних продуктів з метою переходу на використання централізованих БД, авторизації. Завершеним результатом має стати цілісна система електронного документообігу. У більш віддаленій перспективі система може бути доповнена геопросторовими модулями та іншими інструментами моніторингу.

Список літератури

1. SSTC NRS expert activity. URL: <https://www.sstc.com.ua/en/activities/expert-activity.html> (дата звернення: 27.09.2019).
2. Mukhopadhyay, S., Baciú, F., Stowisek, J., Saluja, G., Kenny, P., Albinet, F., International Radiation Monitoring and Information System (IRMIS), Proc. SPIE Radiation Detectors in Medicine, Industry, and National Security XVIII, 10393, 1039305 (2017). doi: 10.1117/12.2275207.
3. Попов О.О., Яцишин А.В., Ковач В.О., Артемчук В.О., Тарадуда Д.В., Соби́на В.О., Соколов Д.Л., Демент М.О., Яцишин Т.М. Концептуальні підходи до створення інформаційно-аналітичної експертної системи для оцінки впливу АЕС на довкілля. Ядерна та радіаційна безпека. 2018. № 3(79). С.56-65.
4. Grossi, P., Souza, L., Figueiredo, G., Figueiredo A. Management information system applied to radiation protection services. 2013 International Nuclear Atlantic Conference - INAC 2013. Recife, PE, Brazil (2013). URL: https://inis.iaea.org/collection/NCLCollectionStore/_Public/46/021/46021431.pdf (дата звернення 20.08.2019)
5. Kashparov, V., Levchuk, S., Zhurba, M., Protsak, V., Khomutinin, Yu., Beresford, N., Chaplow, J. (2018) Spatial datasets of radionuclide contamination in the Ukrainian Chernobyl Exclusion Zone. Earth System Science Data. 10, pp. 339–353, doi: 10.5194/essd-10-339-2018
6. Ткаченко О.М. (2013). Геопросторова складова інформаційно-аналітичної системи у галузі рослинництва. Науковий вісник Національного університету біоресурсів і природокористування України. Серія: Техніка та енергетика АПК. 2013. № 184. С. 150-157.

References

1. SSTC NRS expert activity. URL: <https://www.sstc.com.ua/en/activities/expert-activity.html> (27.09.2019)
2. Mukhopadhyay, S., Baciú, F., Stowisek, J., Saluja, G., Kenny, P., Albinet, F., International Radiation Monitoring and Information System (IRMIS), Proc. SPIE Radiation Detectors in Medicine, Industry, and National Security XVIII, 10393, 1039305 (2017). doi: 10.1117/12.2275207.

3. O. Popov, A. Iatsyshyn, V. Kovach, V. Artemchuk, D. Taraduda, V. Sobyna, D. Sokolov, M. Dement, T. Yatsyshyn. (2018) Conceptual Approaches for Development of Informational and Analytical Expert System for Assessing the NPP impact on the Environment. Nuclear and Radiation Safety, 3(79), 56-65.

4. Grossi, P., Souza, L., Figueiredo, G., Figueiredo A. Management information system applied to radiation protection services. 2013 International Nuclear Atlantic Conference - INAC 2013. Recife, PE, Brazil (2013). URL: https://inis.iaea.org/collection/NCLCollectionStore/_Public/46/021/46021431.pdf (20.08.2019)

5. Kashparov, V., Levchuk, S., Zhurba, M., Protsak, V., Khomutinin, Yu., Beresford, N., Chaplow, J. (2018) Spatial datasets of radionuclide contamination in the Ukrainian Chernobyl Exclusion Zone. Earth System Science Data. 10. doi: 10.5194/essd-10-339-2018

6. Tkachenko, O. (2013). Geoprostorova skladova informatsiino-analitychnoi systemy u haluzi roslinnytstva [Geospatial component of the information-analytical system in crop]. Scientific Journal of National University of Life and Environmental Sciences of Ukraine. Series "Tekhnika ta energetyka APK", 184, 150-157.

ИНФОРМАЦИОННАЯ СИСТЕМА ЭКСПЕРТИЗ РАДИАЦИОННОЙ БЕЗОПАСНОСТИ

А. Н. Ткаченко, В. Н. Куценко

Аннотация. *Радиационная безопасность является одной из важных составляющих экологической безопасности в целом. Радиационный контроль необходим не только в контексте производства и использования ядерного оружия, но и для стран, имеющих развитую инфраструктуру атомной энергетики или пострадавших от радиационных выбросов (Украина, Белоруссия, Япония и другие). Процесс управления в этой сфере достаточно наукоемок и, разумеется, требует тщательного учета всех разрешительных документов, заключений и отчетности.*

В Украине функционирует Государственный научно-технический центр по ядерной и радиационной безопасности. Он осуществляет научно-исследовательскую деятельность, готовит нормативные документы, экспертные заключения в указанной сфере. На предприятии работала устаревшая информационная система, предназначенная для учета, контроля сроков и состояния выполнения экспертиз и договоров, подготовки отчетности.

Целью исследования является выявление недостатков существующей системы учета экспертиз ГНТЦ ЯРБ, проектирование и реализация программного обеспечения, которое решит проблемы в существующей системе учета экспертиз в сфере ядерной и радиационной безопасности.

В этой статье предлагается описание технологий модернизации этой системы, в частности, предложено обновленную структуру, расширенный функционал, программную реализацию отдельных модулей.

Программное, аппаратное обеспечение и отдельные программные решения в рамках задач соответствуют стандартам жизненного цикла разработки

программного обеспечения, организации баз данных, верификации и документирования. Инструментальной платформой разработки является .NET и язык C #.

Ключевые слова: *радиационная безопасность, система, экспертиза, учет*

RADIATION SAFETY EXPERTISES INFORMATION SYSTEM

O. Tkachenko, V. Kutsenko

Abstract. *Radiation safety is one of the important components of environmental safety in general. Radiation monitoring is necessary not only in the context of the production and use of nuclear weapons, but also for countries with developed nuclear energy infrastructure or affected by radiation emissions (Ukraine, Belarus, Japan etc.). The management process in this area is quite knowledge-based and, of course, requires careful consideration of all permits, conclusions and reporting.*

The State Scientific and Technical Center for Nuclear and Radiation Safety operates in Ukraine. He carries out research activities, prepares regulatory documents, expert opinions in this area. The company had an outdated information system designed to record, control the timing and status of the implementation of examinations and contracts, and prepare reports.

The purpose of the study is to identify the shortcomings of the existing NSTC NRS examination system, design and implement software that will solve problems in the existing nuclear and radiation safety expertise system.

This article provides a description of the modernization technologies of this system. In particular, an updated structure, advanced functionality, and software implementation of individual modules are proposed.

The software, hardware, and individual software solutions within the tasks meet the standards of software development, database organization, verification, and documentation. The development toolkit is .NET and the C # language.

Key words: *radiation, safety, system, expertise, accounting*