

МЕТОД СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПУЛЬТА ІНСТРУКТОРА АВІАЦІЙНОГО ТРЕНАЖЕРУ

Пропонується метод для створення програмного забезпечення візуального інтерфейсу пульта інструктора авіаційного тренажеру. Метод базується на понятті ролі при формуванні інтерфейсу, шаблонному проектуванні при створенні програмного забезпечення пульта.

The method for creating software visual interface instructor operating stations of flight simulator is proposed. The method on the concept of interface design role, and patterns design for software development is based.

Вступ

Авіаційна техніка – коштовна складова транспортної галузі та Збройних сил України. Внаслідок обмеженого фінансування та кризових явищ зараз є значні труднощі щодо розробки та впровадження нової техніки, тому проблема підтримки придатності авіаційної техніки є актуальною [1, 2].

Одним із основних елементів авіаційної техніки є авіаційний тренажер [3]. Використання авіаційних тренажерів для зменшення вартості льотної підготовки, покращення екології набуває особливої актуальності в умовах економічної кризи країни. В зв'язку з цим, зараз до авіаційних тренажерів висуваються більш жорсткі вимоги щодо гнучкості, мобільності та вартісних складових.

Невід'ємною частиною авіаційного тренажера є пульт інструктора, який відображає інформацію про політ та стан тренажеру (рис. 1). Існуючі пульти більшості успадкованих тренажерів [4] реалізовані апаратно, морально застарілі та фізично зношені в процесі тривалої експлуатації і, найчастіше не підлягають ремонту із-за припинення виробництва їх компонентів і запасних частин. Це є причиною заміни апаратного пульта інструктора на нові апаратно-програмні засоби на основі сучасних комп'ютерних технологій [5].

В якості апаратного забезпечення використовуються персональні або промислові комп'ютери загального призначення. Програмне забезпечення пульта включає системне програмне забезпечення (операційні системи, протоколи обміну, драйвери) і прикладне програмне забезпечення, яке реалізує функціональність пульта [6].

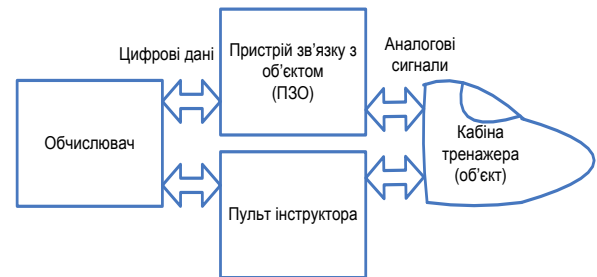


Рис. 1. Склад авіаційного тренажеру

При розробці програмного забезпечення пульта інструктора авіаційного тренажера створення програмного забезпечення візуального інтерфейсу є однією із основних задач. Для вирішення цієї задачі в статті пропонується метод, сутність якого полягає в використанні поняття ролі при формуванні інтерфейсу, представленні інтерфейсних елементів як композиції елементів та шаблонному проектуванні при створенні програмного забезпечення пульта.

Аналіз існуючих підходів щодо створення пультів інструктора авіаційного тренажера

Навчання пілотів на тренажері управляється і контролюється інструктором, в обов'язки якого входить наступне: установка початкових умов польоту, контроль якості пілотування, імітація взаємодії пілотів з диспетчером, управління імітацією відмов літака, розбір помилок і особливостей пілотування (рис. 2) [7]. Для оцінки якості пілотування і розбору польотів інструктор отримує значення відповідних параметрів з фіксацією їх на паперовому чи електронному носіях.

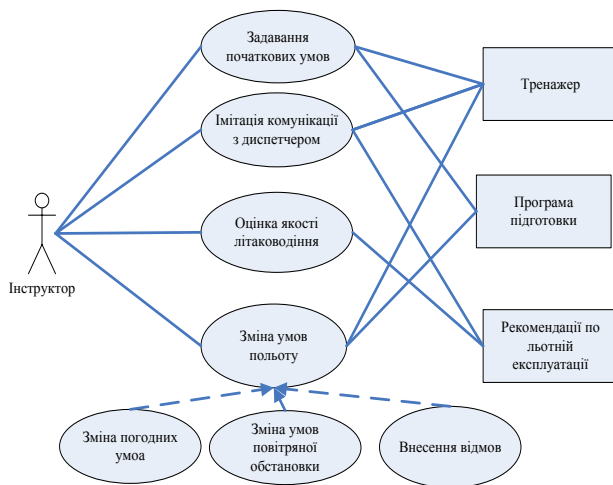


Рис. 2. Функції інструктора

Навчання пілотів відбувається відповідно до програм підготовки, які формуються інструкторами в залежності від конкретних задач навчання, і передбачають проведення серійних учбових польотів на тренажерах в заданих умовах. Програми підготовки можуть змінюватися в період експлуатації літака з врахуванням набутого досвіду пілотування. Тому, для ефективного виконання навчальних дій та швидкого аналізу даних інструктору необхідно володіти детальною інформацією, представленою в доступному для огляду виді [3].

На даний час застосовуються, як тренажери покоління 60-80-х років, так і сучасні тренажери. В процесі експлуатації тренажерів розробки 60-80-х років, уточнити або скоригувати перелік необхідних параметрів дуже складно. До того ж, інформація, що виводиться на індикатори і види управляючих впливів, що здійснюються з пульта жорстко (апаратно) фіксуються при розробці тренажера [3]. Крім цього, такі пульти незручні у роботі через те, що введення відмов та початкових умов польоту здійснюється вручну за допомогою кнопок, ручок та тумблерів на пультах, а вся інформація надається інструктору одночасно, без врахування того, потрібна вона йому в процесі відпрацювання конкретної справи чи ні. Це призводить до того, що інструктор змушений досить багато часу витратити на багатокрокові процедури введення відповідних умов польоту та на відбір інформації, яка йому необхідна на той чи інший момент часу. Отже, внаслідок обмежених технічних можливостей в ергономічному сенсі діяльність інструктора організо-

вана не оптимально. Засоби відображення інформації належним чином не пристосовані до його функцій, а будь-які пристрої для обробки цієї інформації відсутні. Тому актуальною є задача принципової зміни інтерфейсу пульта інструктора [8].

Сучасні пульти інструктора (SLZ-242, Ми-8/Ми-17) реалізуються як апаратно-програмні системи на основі комп'ютерів, що включають засоби візуалізації інформації інтерактивної взаємодії з користувачем і інформаційного обміну з іншими компонентами тренажера. Використання таких систем дозволяє відмовитися від дорогих спеціалізованих пристроїв відображення, вводу і фіксації інформації, що застосовувалися раніше (спеціалізовані індикатори, карти-планшети) і використати комп'ютерні і програмні компоненти [5]. Такі пульти інструктора складаються з одного чи декількох моніторів, миші, клавіатури та принтера. На моніторах за допомогою спеціалізованого програмного забезпечення імітується візуалізація інтерфейсу пульта інструктора (формується зображення індикаторів приладової дошки і відтворюється їх функціонування). До сучасного спеціалізованого програмного забезпечення тренажерів, яке може використовуватися при створенні інтерфейсу пульта інструктора відносять [9]: системи автоматизованого проектування Pro/ENGINEER Wildfire, DeskSim, Segambis.

Інформація на моніторах представляється інтерфейсом виду «вказати-натиснути» (вказати об'єкт на екрані і обрати відривним натиском кнопки миші). За допомогою таких пультів інструктор може швидко створювати звичайні та незвичайні стани в авіаційному тренажері (ланцюжок із загорання двигуна, збої в роботі механізму вивантаження, електричні дефекти, слизька злітно-посадкова смуга, збої в навігаційній системі), зміна погоди в процесі польоту (буря, вітер, блискавка), задання пори року та часу доби, виконання польотів при реальній погоді, якщо комп'ютер підключений до Інтернет, миттєве переміщення літака в будь-яку точку (етап) польоту, переміщення повітряного судна в будь-яку точку земної кулі.

Підхід до створення інтерфейсу пульта інструктора

Зараз, для створення сучасних пультів інструкторів використовують підхід – «фіксо-

вана функціональність», сутність якого полягає в тому, що склад і функціональність пульта визначаються на основі відомих типових задач, що виконуються інструктором при навчанні пілотів [10]. Основним недоліком такого підходу є те, що після створення пульта, види і форми інформації, що відображається та вводиться не змінюються. Для внесення змін необхідна наявність розробника. Тобто, кожен елемент, що міститься на пульті інструктора, розташовується у чітко визначеному місці і не може бути видалений у разі, якщо він не потрібний за даних умов (наприклад, при відпрацюванні певної польотної ситуації).

В роботі [10] пропонується апробований підхід для подолання зазначеного недоліку. Кожен інтерфейсний елемент інтерфейсу пульта інструктора тренажера (кнопка, індикатор тощо) реалізований у вигляді окремого програмного модулю, який динамічно підключається, та представляє собою .Net-збірку, що реалізує певну функціональність. Це дозволяє з окремих інтерфейсних елементів збирати певний інтерфейс [11].

Як розвиток зазначеного підходу для підвищення гнучкості побудови інтерфейсних елементів та настроювання інтерфейсу під різні види задач в статті пропонується застосування шаблонного проектування [12] і узагальнення інформації на основі поняття ролі.

В авіаційному прикладному домені роль визначається як сукупність функціональних обов'язків, властивих авіаційному персоналу. Авіаційний персонал – це особовий склад авіаційних підприємств, організацій, підрозділів і учбових закладів, що складаються із спеціалістів за відповідними професійними ознаками (члени екіпажу повітряного судна, командно-управляючий, інструкторський, диспетчерський склад і т.п.) [4, 13]. Для кожного спеціаліста визначені типові посадові функціональні обов'язки, вимоги до знань і кваліфікації, тобто його роль [14].

В процесі своєї діяльності авіаційний спеціаліст отримує і забезпечує визначену інформацію через технічні системи. Сукупність видів інформації і способів її відображення, властивий одній авіаційній ролі будемо називати рольовим представленням. Таку інформацію можна представити як набір окремих інформаційних елементів, які групуються в логічно і ергономічно пов'язані групи. Ос-

танні будемо називати панелями представлень. Наприклад, до панелей представлення інструктора можна віднести навігаційну приладову панель, панель управління силовою установкою, навколишнє візуальне оточення кабіни літака (рис. 3).

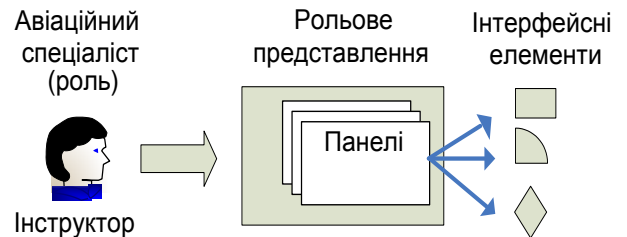


Рис. 3. Роль, панелі представлення і інтерфейсні елементи

Індикатори і органи управління пульта тренажера складають користувальницький інтерфейс інструктора, через який він отримує інформацію про політ і управляє тренажером. Кожний індикатор відображає один параметр польоту. Множина індикаторів, доступних інструктору складає реалізацію його рольового представлення і повинна визначатися підмножиною всіх параметрів польоту, і способами їх представлення, прийнятими в авіації, з врахуванням конкретного типу літака. Органи управління пульта визначаються потребами управління тренажером.

Набір і вид інтерфейсних елементів індивідуальні для тренажера літака кожного типу, однак суттєво схожі. Кожний інтерфейсний елемент $ie = (it, pf)$ визначається двома складовими – видом інформації (параметром) it і формою її представлення pf . Рольове представлення rv можна описати як множину інтерфейсних елементів, що входять в нього $RV = \{ie_i \mid ie_i \in RV\}$, а роль R в свою чергу – як множину панелей представлення $R = \{RV \mid RV \in R\}$. Вся множина інтерфейсних елементів домену $IE = \{ie_i \mid ie_i \in IE, i = 1..d\}$ визначається як множина двійок $ie = (it, pf)$ всіх його підмножин IE_r , визначених для ролей: $R(IE) = \{IE_r \mid IE_r \subseteq IE\}$.

Інтерфейс для ролі інструктора синтезується шляхом вибору та настройки визначених інтерфейсних елементів:

$$r_{instr} = RV_{instr} \subseteq \{ie_i = (it, pf) \mid i = 1..d\}.$$

Створення інтерфейсних елементів

Інтерфейсні елементи пульта інструктора за призначенням умовно поділяють на два класи управляючі – для вводу інформації

(перемикачі режимів, регулятори, кнопки вводу значень) та індикатори, призначені для її відображення (індикатори швидкості, висоти, зображення карт). В свою чергу, індикатори приймають вигляд шкали-стрілки, цифрових табло та візирів [15].

Кожний інтерфейсний елемент можна розглядати як композицію простих елементів: стрілок, шкал, діапазонів величин, маркування, кожний із яких характеризується певними властивостями і поведінкою (рис. 4).

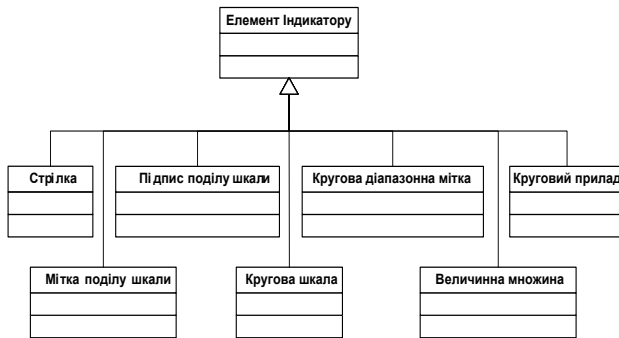


Рис. 4. Композиція елементів індикатору

Елементи індикатору представляються класами, що характеризуються певними властивостями. Атрибути класу відображають властивості елементів, які можуть бути спільними (наприклад, колір, радіус, діапазон величин), і індивідуальними (наприклад, довжина стрілки).

Для побудови індикаторів різного виду був використаний шаблон проектування «компонувальник» (composite) [16]. Він об'єднує об'єкти в деревоподібну структуру для представлення ієрархії від окремого до цілого та дозволяє клієнтам звертатися до окремих об'єктів і до їх груп однаково. На рис. 5 (а, б) представлена UML діаграма шаблону і відповідно, окремий випадок індикаторного приладу. GaugeElement (індикатор) є абстрактним класом, що включає дочірні елементи Pointer (стрілка) та CircularScale (кругова шкала). Даний шаблон дає можливість здійснювати звернення до функціональності індикатору в цілому, а також до простих і складних елементів однаково. Так, метод прорисовки Render викликає послідовно цей метод у всіх елементів композиції (Pointer, CircularScale), проходячи по ієрархії класів.

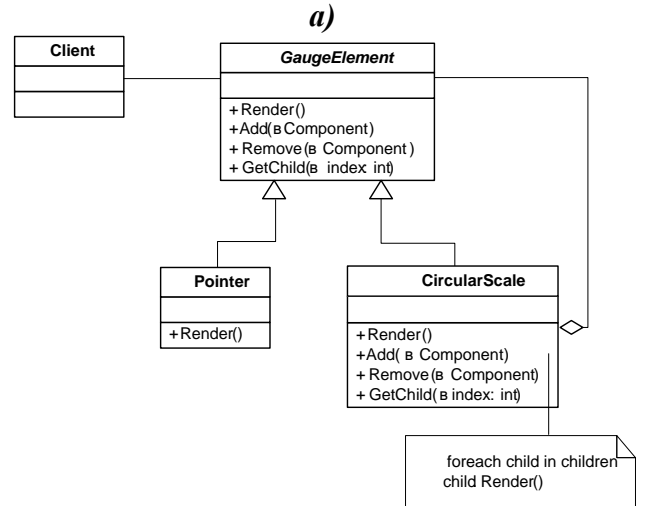
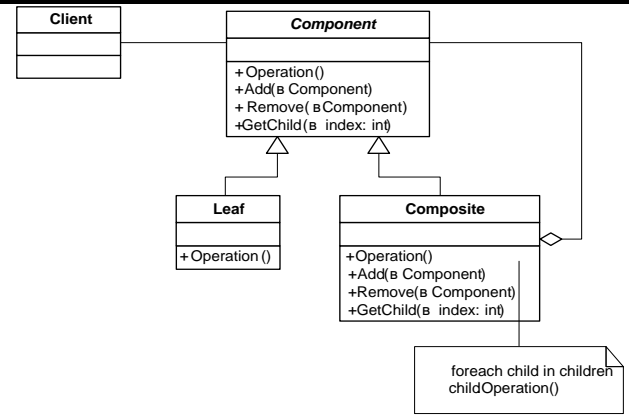


Рис. 5. UML діаграма шаблону «компонувальник» для індикаторів

В запропонованому підході індикатори будуються як композиції елементів (наприклад, круговий прилад включає в себе кругову шкалу, стрілку, мітку та підпис поділу шкали) (рис. 6).

Для того, щоб створити певний інтерфейсний елемент необхідно визначити склад елементів і задати значення їх атрибутів.

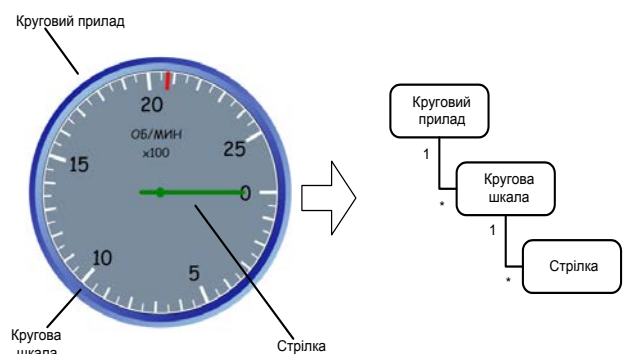


Рис. 6. Представлення інтерфейсного Елемента

Реалізація бібліотеки класів

На основі класифікації і специфікування властивостей елементів в роботі побудована ієрархія класів, на базі якої розроблена бібліотека. Шаблон «компонувальник», викори-

стовується як базова архітектурна концепція ієрархії (рис. 7).

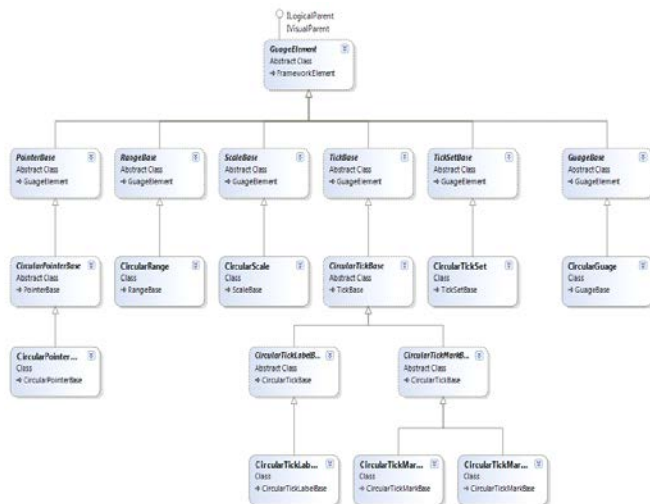


Рис. 7. Архітектура бібліотеки класів індикатору

Всі класи бібліотеки наслідуються від базового класу GaugeElement, який має можливість композиції собі подібних елементів.

Бібліотека класів реалізована з використанням технології Windows Presentation Foundation, яка входить в склад .NET Framework 3.0. Для створення інтерфейсних елементів пульта застосовується мова розмітки XAML, що входить до складу Windows Presentation Foundation. Використання Windows Presentation Foundation сумісно з XAML дає можливість змінювати масштабованість індикаторів без втрати якості зображення, динамічно створювати нові індикатори і їх оформлення. Розробник може будувати індикатори відповідного типу, включаючи до композиції різноманітні елементи із бібліотеки.

Створюваний індикатор має деревоподібну структуру піделементів. Наприклад, структура приладу «тахометр» (рис. 8):

1. Круговий прилад (Circular Gauge);
 - 1.1 Кругова шкала (Circular Scale);
 - 1.1.1. Величинна множина (Tick Set);
 - 1.1.1.1. Мітка поділу шкали (Tick Mark);
 - 1.1.1.2. Підпис поділу шкали (Tick Label);
 - 1.1.1.3. Стрілка(Pointer);

Використання XAML робить створення приладів більш зрозумілим і відповідно швидшим, порівняно з застосуванням С# [15].

```
<ctrls:CircularGauge Radius="100" >
<ctrls:CircularGauge.Scales>
<ctrls:CircularScale Radius="99" StartAngle="-225" SweepAngle="55" >
<ctrls:CircularScale.TickSets>
<ctrls:CircularTickSet Minimum="0" Maximum="5.25" >
<ctrls:CircularTickSet.Ticks>
<ctrls:CircularTickMarkMajor ScalePlacement="Inside" TickMarkType="Rectangle" />
<ctrls:CircularTickLabelMajor StartValue="0" EndValue="5.2" ValueSign="Positive"
Interval="1" ScalePlacement="Inside" />
<ctrls:CircularTickSet.Ticks>
<ctrls:CircularTickSet.Pointers>
<ctrls:CircularPointerNeedle Background="Green" Value="{Binding ElementName=UcElement,
Path=Value} />
</ctrls:CircularTickSet.Pointers>
</ctrls:CircularTickSet>
</ctrls:CircularScale.TickSets>
</ctrls:CircularScale>
</ctrls:CircularGauge.Scales>
</ctrls:CircularGauge>
```

Рис. 8. Створення приладу за допомогою мови XAML

ВИСНОВОК

Визначено задачу вдосконалення технології створення програмного забезпечення пультів інструкторів авіаційних тренажерів в напрямку створення можливостей адаптації інтерфейсу до особливостей навчально-тренувального процесу.

Для створення візуального інтерфейсу пульта інструктора запропоновано метод, особливістю якого є можливість динамічної адаптації безпосередньо в процесі експлуатації, що дозволяє підвищити ефективність використання комп'ютерних авіаційних тренажерів в навчально-тренувальному процесі. Метод базується на використанні поняття ролі при формуванні інтерфейсу, класифікації інтерфейсних елементів, специфікуванні їх властивостей та побудові програмного середовища за допомогою шаблону проектування «компонувальник» (composite).

Запропонований метод реалізовано у вигляді бібліотеки класів, на основі якої побудовано гнучку систему індикаторів пульта інструктора тренажеру ТЛ-410М. Аналіз практичного використання такого пульта показав ефективність запропонованого підходу з точки зору якості підготовки авіаційних спеціалістів та перспективність подальших розробок у цьому напрямку.

Список посилань

1. Луцький М.Г. Підтримка придатності програмного забезпечення при модернізації та створенні авіаційної техніки // Матеріали науково-технічної конф. «Створення та модернізація озброєння і військової техніки МОУ в сучасних умовах». – Ф. – 2009.

2. Луцький М.Г., Сидоров М.О., Зіатдінов Ю.К. Підтримка придатності інформаційно-програмного забезпечення авіаційної техніки // Матеріали науково-практичної конф. «Актуальні проблеми розвитку авіаційної техніки». – К. – 2009.
3. Меерович Г.Ш., Годунов А.И., Ермолов О.К. Авиационные тренажеры и безопасность полетов. – М.: Воздушный транспорт. 1990. – 343 с.
4. Сидоров Н.А., Недоводеев В.Т., Хоменко В.А., Сердюк И.П., Сидоров Е.Н. Реинженерия программного обеспечения информационно-моделирующих тренажерных комплексов. // Управляющие системы и машины. – К. - 2008. – № 4. – С.68-74.
5. Design of a flight simulator software architecture. Göran Ancker, Jan Wallenberg. – School of Mathematics and Systems Engineering, Växjö University. – 2002. – 91 p.
6. Сидоров М.О., Иванова Л.М., Хоменко В.А. Методологічні принципи реінженерії програмного забезпечення успадкованих авіаційних тренажерів // Матеріали VIII Міжн. наук-техн. конф. АВІА-2007 – К. – 2007. – Т.1. – с. 13.119 – 13.122.
7. Хоменко В.А., Сидоров Є.М. Повторно используемое решение для программного обеспечения пульта инструктора авиационного тренажера// Матеріали науково-практичної конф. «Актуальні проблеми розвитку авіаційної техніки». – К. – 2009. – с. 118.
8. Іваськевич І. О. Ергономіка: Навчальний посібник // - Т.: «Економічна думка», 2002. –165 с.
9. Желонкин В.И. Система поддержки исследований по выбору и оптимизации видов электронной индикации // Вестник Международной Академии проблем человека в авиации и космонавтике. – М.– 2007. №3(26). – с. 33-39.
10. Сидоров Е.Н. Архитектура программного обеспечения пульта инструктора авиационного тренажера // Матеріали міжнародної науково-практичної конф. аспірантів і студентів «Інженерія програмного забезпечення 2008». – К. – 2009. – с. 45-49.
11. Хоменко В.А., Сидоров Є.М., Артєєв Т. КП «Програмне забезпечення робочого місця інструктора успадкованого авіаційного тренажера ТЛ-410М». Авторське свідоцтво № 28163.
12. Хоменко В.А, Сидоров Е.Н, Мендзєбровський І.Б. Шаблон программного обеспечения устройств связи с объектом авиационных тренажеров. // Проблемы программирования. – К. – 2008. – С. 30 –40.
13. Повітряний кодекс України. Розділ 5. – К. – 1993.
14. Сидоров М.О. Групова динаміка і комунікації. – К.:НАУ, 2009. – 87с.
15. Хоменко В.А., Авраменко Е.А., Рябоконт Ю.Н. Подход к созданию визуального интерфейса пульта инструктора авиационного тренажера. // Матеріали науково-практичної конф. «Актуальні проблеми розвитку авіаційної техніки». – К. – 2009. – с. 117.
16. Ларман К. Применение UML и шаблонов проектирования. – М.:«Вильямс», 2002. – 624 с.

Поступила в редакцию 16.12.2009