

# Применения (опыт разработки и внедрения информационных технологий)

---

DOI <https://doi.org/10.15407/usim.2018.03.051>

УДК 004.9

**В.В. ЗОСИМОВ**, канд. техн. наук, доцент,  
zosimovvv@gmail.com

**О.В. ХРИСТОДОРОВ**, магістр спец. 122 «Комп'ютерні науки»,  
belfegor26@gmail.com

**О.С. БУЛГАКОВА**, канд. техн. наук, доцент,  
sashabulgakova2@gmail.com

Кафедра комп'ютерних наук та прикладної математики Миколаївського національного університету ім. В.А. Сухомлинського, м. Миколаїв, 54030, вул. Нікольська, 24

## Технологія автоматизованої адаптації веб-додатків на основі ідентифікації кіберсутностей

---

*Описано програмне забезпечення для автоматизованої адаптації користувацьких інтерфейсів. Подано стислий опис програмної розробки, спрямованої на автоматизовану адаптацію інтерфейсів під потреби користувачів. Програмний продукт забезпечує псевдоідентифікацію користувачів (вибудування бази анонімних користувачів та правил на основі їх перебування у веб-додатках).*

**Ключові слова:** Інтерфейс, адаптивність, індуктивне моделювання, веб-інтерфейс.

### Вступ

Інтерфейси — невід'ємна складова ланок сприйняття та управління програмними продуктами. Саме вони забезпечують ці функції та зв'язок користувачів з програмою. Невпинна інтеграція ІТ технологій у існування сучасної людини створює зростаючий попит на розробки у галузі адаптації інтерфейсів програмних продуктів, спрямованих на потреби користувачів. До суттєвих переваг використання цих продуктів, скерованих на підвищення зручності користування інтерфейсами, можна віднести: спрощення сприйняття бізнес-логіки програмного продукту конкретному користувачу; гнучкість моделі представлення інформації; підвищення продуктивності роботи з інтерфейсом. Під динамічною змі-

ною слід розуміти зміну відображення (адаптацію) інтерфейсу як результат виконання деякого сценарію на основі поведінкового портрету користувача.

Найбільш повний огляд вмісту адаптивної поведінки ЛМІ (людино-машинного інтерфейсу) маємо в [1]. Автори пропонують три параметра інтерфейсу, які можуть змінюватися:

- зміст інформації, що представляється;
- форма подання інформації та ведення діалогу;
- розподіл завдань між людиною і машиною (рівень автоматизації).

В інших роботах декларується, що адаптивність інтерфейсу проявляється:

- в налаштуванні рівня деталізації діалогу з користувачем — від докладного, «провідного»

користувача до мети крок за кроком через ієрархію меню, до короткого, з використанням скорочених команд і макросів в режимі «питання—відповідь» [2];

- в підказках, обмеження доступу до додатків, регулювання інтенсивності інформаційного обміну і зміні зовнішнього вигляду інтерфейсу [3, 4];

- у фільтрації та розстановці пріоритетів контенту, запропонованого користувачеві, що може відбуватися не тільки відповідно до власних уподобань, а й під впливом зовнішніх факторів і контексту (наприклад, користувачеві пред'являється продукт, який вигідно купити саме зараз [5]);

- у зміні темпу подачі інформації [6];

- в налаштуванні параметрів зображення (таких як товщина ліній, розмір шрифту, яскравість та ін.).

За умов конкуренції сучасних розробок у сферах використання інтерактивних інтерфейсів обумовлюється актуальність досліджень питань реалізації автоматизованої адаптації користувацьких інтерфейсів. Під автоматизованою адаптацією слід розуміти динамічну зміну користувацьких інтерфейсів на основі автоматично зібраної інформації про користувача.

Невпинний розвиток галузі ІТ технологій породжує попит на еволюцію досліджень моделей та методів автоматизованої адаптації інтерфейсів. Значну роль в дослідженні моделей та методів адаптації інтерфейсів відіграють досягнення, висвітлені у [7]. Окремі питання проектування і аналізу використання можливостей псевдоідентифікації розглянуто у працях [8, 9]. Деякі питання використання технологій збору даних у автономних кіберфізичних системах розглянуто у працях вітчизняних науковців [10]. В [8–10] дослідниками вивчено фундаментальні принципи та концепції, які застосовуються при реалізації програмних продуктів; представлені у [8] різні моделі та методи, покликані покращувати сприйняття та взаємодію з інтерфейсами кінцевих програмних продуктів. Але всі ці аспекти розглядалися як окремі задачі, та не було їх поєднання в адаптивну систему.

Глибокий аналіз досліджень дозволив дійти висновку, що на сьогодні невпинно росте попит на системи адаптації інтерфейсів у всіх сферах їх використання. Автоматично адаптований інтерфейс має чималу кількість стратегічних переваг серед конкурентоспроможних інтерфейсів аналогічної спрямованості.

## Індуктивний підхід до побудови адаптивних інтерфейсів

Нехай  $CR = \{cr_1, \dots, cr_{|CR|}\}$  — множина критеріїв, які будуть змінюватися, тобто адаптуватися під користувача,  $Fact = \{fact_1, \dots, fact_{|Fact|}\}$  — це множина факторів, які впливатимуть на вибір критерію.

На основі виділених заздалегідь факторів, які можуть впливати на інтерфейс, будуть змінюватися критерії побудови інтерфейсу. Індуктивність процесу полягає в тому, що адаптація інтерфейсу відбувається від конкретних даних спостережень, тобто факторів — до загальної моделі [11], яка містить множину критеріїв, що змінюватимуться (наприклад, з відомої інформації про вік користувача (вік — фактор), можна збільшити розмір шрифту (для похилого віку) — критерій).

Отже, виходячи зі сказаного, можна побудувати функцію  $\Phi$ , яка відобразить процес адаптації.

$$\begin{aligned} \Phi(cr_1(fact_1, \dots, fact_k), \dots, cr_m(fact_k, \dots, fact_F)) = \\ = \theta_0 + \sum_{i=1}^m \theta_i cr_i \sum_{F=1}^k (fact_1, \dots, fact_F) + \\ + \sum_{i=1}^m \sum_{j=1}^m \theta_{ij} cr_i cr_j \sum_{F=1}^k (fact_1, \dots, fact_F) + \\ + \sum_{i=1}^m \sum_{j=1}^m \sum_{k=1}^m \theta_{ijk} cr_i cr_j cr_k \sum_{F=1}^k (fact_1, \dots, fact_F) + \dots \end{aligned}$$

Загальна функціональна схема механізму адаптації інтерфейсу (рис. 1) складається з декількох блоків: блок визначення факторів, в якому формуються/зберігаються фактори користувача, які можуть впливати на критерії зміни інтерфейсу, тобто адаптації; блок формування критеріїв (кількість критеріїв для кожного фак-

тора може бути різною); блок обробки даних, в якому програмно оброблятиметься вміст критеріїв, що призведе до зміни інтерфейсу [12].

### Вимоги до програмного продукту

Програмний продукт має забезпечувати: збір та зберігання інформації щодо користувачів застосованого веб-додатку; на підставі зібраних даних псевдоідентифікувати користувачів; автоматизовану адаптацію інтерфейсів псевдоідентифікованих користувачів.

Програмний продукт має відповідати таким вимогам:

1. Контрольованому збору інформації щодо користувачів веб-додатку;
2. Створенню та підтримці бази даних отриманої інформації;
3. Псевдоідентифікації користувачів веб-додатку на основі зібраної інформації;
4. Автоматизованій адаптації користувацького інтерфейсу веб-додатку на підставі зібраних та опрацьованих даних програмним продуктом.

Далі проведено аналіз пред'явлених вимог у порівнянні з існуючими аналогами. За останні роки сформувався і продовжує розвиватись напрям коригування текстового наповнення, структури сайту, контроль внутрішніх та зовнішніх чинників перебування та взаємодії користувача з інтерфейсом. Це невід'ємна складова успіху маркетингових та інформаційних сайтів. Існує велика кількість систем збору інформації про користувачів, наприклад, *Amplitude*, *Mixpanel* та ін.

Представлені програмні продукти задовольняють вимоги 1 та 2, але не задовольняють вимог 3 та 4. У зв'язку з цим створення систем автоматизованої адаптації програмних продуктів є актуальною.

Також величезна кількість систем, покладених на аналіз поведінки користувачів, та практична відсутність систем динамічної адаптації інтерфейсів на базі зібраної аналітичної інформації щодо поведінкового використання контенту, викликало потребу у створенні програмного продукту даного спрямування.

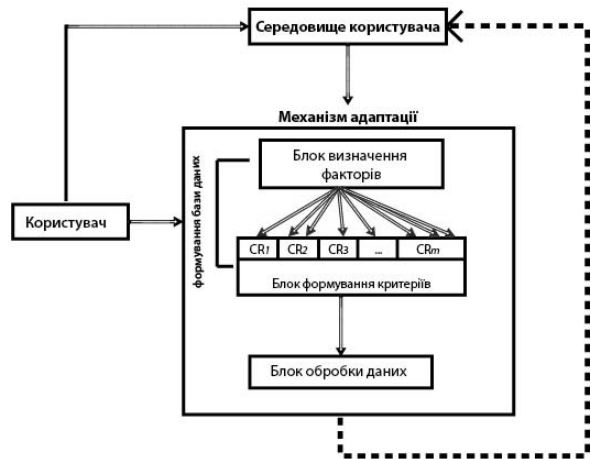


Рис. 1. Складові механізму адаптації інтерфейсу

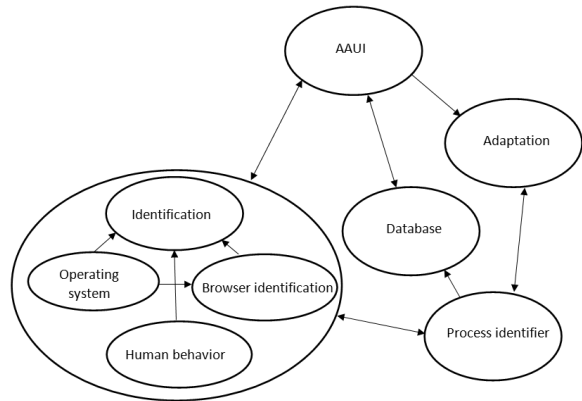


Рис. 2. Схема взаємодії програмної системи «AAUI»

### Структурна схема програмного продукту

Схема взаємодії програмного комплексу представлена на рис. 2.

Блок *AAUI* (*Automatic Adaptation User Interface*) є початковим модулем, з якого відбувається взаємодія з компонентами програмного продукту, *Identification* — реалізує формування сутності з зібраної інформації про об'єкт, що проявив активність, *Operating system* — збір інформації про програмні засоби активного об'єкта відвідування, *Browser identification* — відповідає за збір інформації про програмне забезпечення, яке застосовується у взаємодії з кінцевим програмним

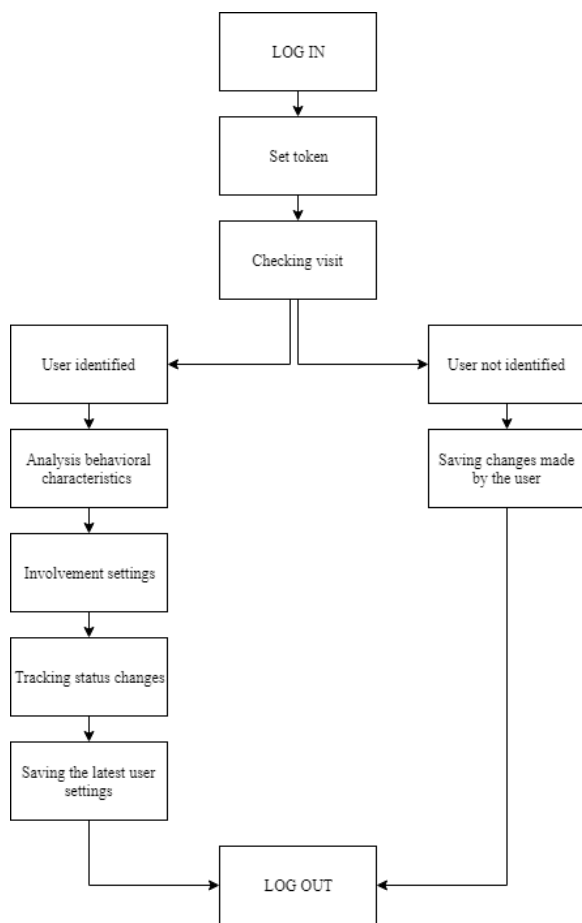


Рис. 3. Схема модуляції прикладного застосування програмного продукту

продуктом, *Human behavior* — забезпечує перевірку можливої минулої присутності псевдодеанімізованого об'єкту, *Adaptation* — блок відповідає за закладену програмну адаптацію, реалізовану на підставі утвореного відбитку псевдодеанімізованого об'єкту взаємодії з кінцевим продуктом *Database* — блок відповідає за збереження проаналізованої інформації, токів псевдодеанімізації та правил адаптації інтерфейсів, *Process identifier* — блок відповідає за узгодження ідентифікації існуючих об'єктів взаємодії.

Далі показано блок-схему модуляції при застосуванні програмного продукту (рис. 3).

Блок *LOG IN* — відображає відвідування користувачем сервісу; блок *Set token* — відобра-

жає генерацію ідентифікатора програмною системою *AAUI*, генерація ідентифікатора відбувається паралельно зі збором додаткових даних, для більш уточненої ідентифікації користувача (рис. 4).

*Checking visit* — перевірка на можливе минуле відвідування користувачем сервісу:

- у разі ідентифікації користувача *User identified* відбувається аналіз *Analysis behavioral characteristics* поведінкових характеристик, зібраних на основі запрограмованих на відстеження змін об'єктів інтерфейсу програмного продукту, застосування персональних налаштувань *Involvement settings* та подальше стеження за зміною потенційно змінюваних налаштувань інтерфейсу *Tracking status changes*. Всі зміни налаштувань веб-інтерфейсу, що відстежуються, зберігаються *Saving the latest user settings* у базу даних на кожному етапі їх здійснення;

- у разі, якщо користувач вперше відвідує веб-сервіс *User not identified*, та збігів ідентифікації не знайдено, відбувається відстеження заданих змін користувачем та подальше збереження *Saving changes made by the user* на етапі їх здійснення.

## Програмна реалізація

Основними перевагами програмної системи є застосована модель ідентифікації анонімних користувачів кінцевих програмних продуктів для подальшого використання динамічного ідентифікатора з метою автоматичної адаптації інтерфейсу під ідентифікованого користувача.

*AAUI* є серверним програмним продуктом, написаним на скриптовій мові програмування PHP.

Програмний продукт забезпечує:

- адміністрування програмного продукту (управління списком застосованих фільтрів ідентифікації, зміна, редагування, створення правил адаптації);
- зберігання та обробка інформації щодо користувачів;
- забезпечення зберігання інформації щодо сесій.

У *AAUI* передбачено механізм налаштування псевдоанімізації користувачів та групування їх за заданими фільтрами.

Функції груп користувачів:

- групування списків за країною відвідування;
- групування списків користувачів за мовою користувачів;
- групування за конфігурацією кінцевого пристрою відвідувача;
- комбінація групувань для виокремлення цільової аудиторії.

Надійність та стійке функціонування *AAUI* досягається сукупністю таких організаційно-технічних заходів:

- організацією надійного захисту спеціалістами з кібербезпеки веб-додатків;
- організацією регулярного і якісного технічного обслуговування серверної частини;
- своєчасним обслуговуванням бази даних.

*Резервне копіювання* БД програмного продукту має виконуватись у терміни, встановлені виробничою необхідністю.

Джерелом даних є база даних. В ній зберігається вся інформація щодо псевдоанімізованих користувачів та правил поведінки з встановленими групами.

Кінцеві користувачі взаємодіють з програмною системою через первинну взаємодію з інтерфейсом (рис. 5).

Серверний додаток *AAUI* взаємодіє з відвідувачем веб-додатку у момент його початку перебування та взаємодії з інтерфейсом останнього. Рівень поглинання зібраної інформації про відвідувача встановлює адміністратор програмної системи.

Основним джерелом зберігання даних є база даних. В ній зберігається інформація щодо отриманих сутностей псевдоанімізованих користувачів.

Для зберігання інформації використовується вільна система управління реляційними базами даних *MySQL*. Структуру бази даних розроблено на основі виробленої методики ідентифікації користувачів.

Для візуального представлення бази даних була використана методологія *IDEFIX*, в заса-

```

171 function _getBrowserFingerprint() {
172     $useragent = $_SERVER['HTTP_USER_AGENT'];
173     $accept    = $_SERVER['HTTP_ACCEPT'];
174     $charset   = $_SERVER['HTTP_ACCEPT_CHARSET'];
175     $encoding  = $_SERVER['HTTP_ACCEPT_ENCODING'];
176     $language  = $_SERVER['HTTP_ACCEPT_LANGUAGE'];
177     $data = '';
178     $data .= $useragent;
179     $data .= $accept;
180     $data .= $charset;
181     $data .= $encoding;
182     $data .= $language;
183
184     $hash = hash('sha256', $data);
185     return $hash;
186 }

```

Рис. 4. Реалізація блоку генерації ідентифікатора



Рис. 5. Схема моделі взаємодії

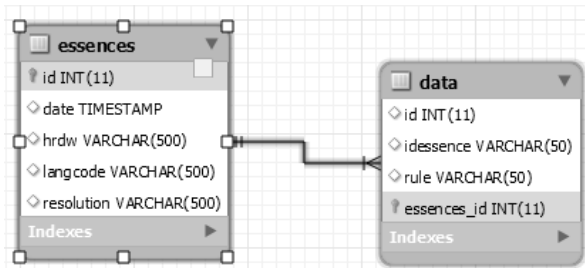


Рис. 6. ER-діаграма бази даних у нотатції *IDEFIX*

дах якої полягає моделювання реляційних баз даних (рис. 6). Зазначений стандарт входить до сімейства методологій *IDEF*, що дозволяють досліджувати структуру, параметри і характеристики виробничо-технічних та організаційно-економічних систем. Методологія *IDEFIX* адаптована для спільного використання з *IDEFO* в межах єдиної технології моделювання. Тобто в межах *IDEFO* деталізуються функціональні блоки, а в рамках *IDEFIX* деталізуються стрілки, які взаємодіють з функціями. Розробка бази даних відбувалась інструментом для проектування баз даних *MySQL Workbench* — інструмента, який інтегрує проектування, моделювання, створення й експлуатацію БД в єдине безкоштовне оточення для

```

118 App::before(function($request){
119     $url_lang = Request::segment(1);
120     $cookie_lang = Cookie::get('language');
121     $browser_lang = substr(Request::server('HTTP_ACCEPT_LANGUAGE'), 0, 2);
122     if(!empty($url_lang) AND in_array($url_lang, Config::get('app.languages')))
123     {
124         if($url_lang != $cookie_lang)
125         {
126             Session::put('language', $url_lang);
127         }
128         App::setLocale($url_lang);
129     }
130     else if(!empty($cookie_lang) AND in_array($cookie_lang, Config::get('app.languages')))
131     {
132         App::setLocale($cookie_lang);
133     }
134     else if(!empty($browser_lang) AND in_array($browser_lang, Config::get('app.languages')))
135     {
136         if($browser_lang != $cookie_lang)
137         {
138             Session::put('language', $browser_lang);
139         }
140         $timezone = \Auth::user()->timezone;
141         $datetime = $this->getDate($value);
142         DB::table('essences')->insert([
143             ['langcode' => $browser_lang, 'votes' => 0]
144             ['date' => $datetime + $timezone, 'votes' => 0]
145         ]);
146     }
147 }
148 }

```

Рис. 7. Блок збору інформації

```

203 public function handle($request, Closure $next)
204 {
205     if(!\Session::has('locale'))
206     {
207         \Session::put('locale', \Config::get('app.locale'));
208     }
209     $usersrule = DB::table('data')->select('rule', 'rule')->get();
210     if(checkrule($usersrule) == Config::get('app.locale'))
211     {
212         app()->setLocale(\Session::get('usersrule'));
213     }
214     return $next($request);
215 }

```

Рис. 8. Функція автоматичної адаптації

системи баз даних *MySQL*, зокрема у нотації *IDEFIX*.

В даному продукті цікава саме реалізація системи псевдоідентифікації користувачів. Взаємодію між частинами програмного продукту розглянемо на прикладі таких частин коду.

Представлено блок отримання інформації щодо встановлених користувачем мов, часового поясу (рис. 7).

Функція автоматичної адаптації локалізації веб-додатку на основі зібраних даних показано на рис. 8.

## Експериментальне застосування технології

Під час інтеграції *AAUI* з Інтернет-магазином, було проведено дослідження, спрямоване на виявлення відсоткового співвідношення приросту кількості нових відвідувачів. Протягом

тридцяти днів система працювала у режимі збору поведінкових сутностей, псевдодеанімізації та формування бази відвідувачів.

Отримано вибірку з 2000 унікальних поведінкових портретів. З них 30 відсотків повертались на сайт Інтернет-магазину в подальшому. На підставі аналізу зібраних даних, було запрограмовано адаптацію функції, яка була найбільше запитана — сортування товарів за ціною (від мінімальної до максимальної). Програмна система *AAUI* протягом тридцяти днів генерувала вибірку поведінкових портретів та при повторному відвідуванні псевдодеанімізованого користувача автоматично адаптувала запрограмовану на відстеження функцію, в залежності від отриманого стану сортування.

В результаті використання *AAUI* відсоток відвідувачів веб-додатку, які знову звернулися до Інтернет-магазину зріс з 30 відсотків до 50.

В подальшому буде проведено більш детальне експериментальне дослідження щодо можливостей запропонованої технології.

## Висновки

Опис програмного забезпечення представлено для автоматизованої адаптації користувацьких інтерфейсів. Аналіз систем, спрямованих на збір інформації щодо користувачів, таких як *Amplitude* та *Mixpanel*, дозволив дійти висновку, що зазначені програмні системи забезпечують лише збір профільованої інформації щодо вхідних користувачів, та не забезпечують автоматизованої адаптації інтерфейсів під потреби користувачів.

Стислий опис програмної розробки *AAUI*, спрямованої на автоматизовану адаптацію інтерфейсів розроблено та подано під потреби користувачів. Програмний продукт забезпечує псевдоідентифікацію користувачів (вибудування бази анонімних користувачів та правил на основі їх перебування у веб-додатках). Основними перевагами програмної системи є застосована модель ідентифікації анонімних користувачів кінцевих програмних продуктів задля подальшого використання динамічного іден-

тифікатора з метою автоматичної адаптації інтерфейсу під ідентифікованого користувача.

Головною перевагою ААUI є відсутність у відкритому доступі аналогічного програмного забезпечення, спрямованого на автоматизовану адаптацію інтерфейсів на основі псевдодеанімізованої інформації щодо користувачів інтерфейсів.

До перспектив подальшої розробки ААUI можна віднести збільшення кількості маркерів ідентифікації задля підвищення вірогідності ідентифікації користувачів, впровадження вбу-

довуваності в системі управління контентом та оптимізацію збитковості даних.

Представлене експериментальне застосування технології на прикладі Інтернет-магазину, в якому протягом 30 днів було зафіксовано 2000 унікальних поведінкових портретів. В результаті експерименту відсоток відвідувачів веб-додатку, які знову звернулися до Інтернет-магазину зріс з 30 відсотків до 50. Даний експеримент демонструє зацікавленість в запропонованій технології не тільки з наукової точки зору, а й з комерційної.

## СПИСОК ЛІТЕРАТУРИ

1. Rothrock L., Koubek R., Fuchs F. et al. Review and reappraisal of adaptive interfaces: toward biologically inspired paradigms, *Theoretical Issues in Ergonomics Science*, 2002, **3**, N 1, p. 47–84.
2. Ходаков В.Е., Ходаков Д.В. Адаптивный пользовательский интерфейс: проблемы построения, *Автоматика. Автоматизация, Электротехнические комплексы и системы*, 2003, № 1 (11), с. 45–57.
3. Курзанцева Л.И. Методика комплексного исследования адаптивного человеко-машинного интерфейса, *Математичні машини і системи*, 2011, № 4, с. 69–77.
4. Курзанцева Л.И. Об адаптивном интеллектуальном интерфейсе «пользователь — система массового применения», *Комп’ютерні засоби, мережі та системи*, 2008, № 7, с. 110–116.
5. Langley P. User modeling in adaptive interfaces, *Proc. of the Seventh Int. Conf. on User Modeling*. Banff, Alberta, Canada, June 20–24, 1999. Springer, 1999, p. 357–370.
6. Karwowski W. A review of human factors challenges of complex adaptive systems: discovering and understanding chaos in human performance, *Human Factors*, 2012, **54**, N 6, p. 983–995.
7. Peter Eckersley, How Unique Is Your Web Browser? Electronic Frontier Foundation, <https://www.eff.org/>
8. Jason Gilmore Easy Laravel 5, J. Gilmore — Leanpub, 2016, 235 с.
9. Kelt Dockins Design Patterns in PHP and Laravel, K. Dokins — Appers, 2017, 45 с.
10. Бочкарьов О., Голембо В. Використання інтелектуальних технологій збору даних у автономних кіберфізичних системах, *Lviv Polytechnic National University Institutional Repository*. Зб. наук. праць, 2015, № 830, с. 7–11.
11. Stepashko V., Bulgakova O., Zosimov V. Construction and research of the generalized iterative GMDH algorithm with active neurons. *Advances in Intelligent Systems and Computing*, Springer, Cham **689**, 2018, p. 492–510.
12. Булакова О.С. Концепція побудови адаптивного інтерфейсу з використанням індуктивного підходу. Індуктивне моделювання складних систем: Зб. праць, **8**, К.: МННЦ ІТС, 2016, с. 73–78

Поступила 26.09.2018

## REFERENCES

1. Rothrock, L., Koubek, R., Fuchs, F. et al., 2002. “Review and reappraisal of adaptive interfaces: toward biologically inspired paradigms”. *Theoretical Issues in Ergonomics Science*, 3 (1), pp. 47–84.
2. Khodakov, V.E., Khodakov, D.V., 2003. “Adaptive user interface: problems of construction”. *Automation. Electrotechnical complexes and systems*, 1 (11), pp. 45–57. (In Russian).
3. Kurzantseva, L.Y., 2011. “Metodyka kompleksnoho yssledovanyia adaptivnoho cheloveko-mashynnoho ynterfeisa”. *Matematychni mashyny i systemy*, 4, pp. 69–77. (In Russian).
4. Kurzantseva, L.Y., 2008. “Ob adaptivnom yntellektualnom ynterfeise “polzovatel - sistema massovoho pryemleniya”. *Kompiuterni zasoby, merezhi ta systemy*, 7, pp. 110–116. (In Russian).
5. Langley, P., 1999. “User modeling in adaptive interfaces”. *Proceedings of the Seventh International Conference on User Modeling*. Banff, Alberta, Canada, June 20–24, 1999. Springer, pp. 357–370.
6. Karwowski, W., 2012. “A review of human factors challenges of complex adaptive systems: discovering and understanding chaos in human performance”. *Human Factors*, 54 (6), pp. 983–995.

7. Eckersley, P. How Unique Is Your Web Browser? Electronic Frontier Foundation <https://www.eff.org/>
8. Jason Gilmore Easy Laravel 5 / J. Gilmore — Leanpub, 2016. 235 p.
9. Kelt Dockins. Design Patterns in PHP and Laravel / K. Dokins - Appers, 2017. — 45 p.
10. Bochkarov, O., Holembo, V., 2015. “Vykorystannia intelektualnykh tekhnolohii zboru danykh u avtonomnykh kiberfizychnykh systemakh”. Lviv Polytechnic National University Institutional Repository Zbirnyk naukovykh prats, 830, pp. 7–11. (In Ukrainian).
11. Stepashko, V., Bulgakova, O., Zosimov, V., 2018. “Construction and research of the generalized iterative GMDH algorithm with active neurons”. Advances in Intelligent Systems and Computing, Springer, v. 689, pp. 492–510.
12. Bulgakova, O.S., 2016. “Kontsepsiia pobudovy adaptivnoho interfeisu z vykorystanniam induktyvnoho pidkhotu”. Induktyvne modeliuвання skladnykh system. Zbirnyk prats, vypusk 8. Kyiv: MNNTS ITS, pp. 73–78. (In Ukrainian).

Received 26.09.2018

*V.V. Zosimov*, Ph.D in Techn.Sciences, Associate Professor  
of the Computer Science and Applied Mathematics Department,  
zosimovvv@gmail.com

*O.V. Khrystodorov*, Master degree of Computer Science,  
belfegor26@gmail.com

*O.S. Bulgakova*, Ph.D in Techn.Sciences, Associate Professor  
of the Computer Science and Applied Mathematics Department,  
sashabulgakova2@gmail.com

Computer Science and Applied Mathematics Department

V.O. Sukhomlynsky Mykolaiv National University, Nikolska str., 24, Mykolaiv, 54030, Ukraine,

#### TECHNOLOGY OF WEB APPLICATIONS BASED ON THE CYBER-ENTITIES IDENTIFICATION

**Introduction.** Interfaces are the integral component of the units of perception and management software. They provide software management and user communication with the program. The relentless integration of IT technologies in the contemporary world creates an increasing demand for software development with adapting interfaces field aimed at the needs of users. Therefore, the actual task is to create an automatically adapted interface to the users needs.

**Purpose.** The purpose of this article is to develop the automated adaptation technology of Web applications based on the cyber-entities identification.

**Methods.** A brief description of the software development aimed at the automatic adaptation of interfaces for the users needs is developed and provided. The software product provides pseudo-identification of users (building anonymous user base and rules based on their presence in web applications). The main advantages of the software system are the model of anonymous users identification of the final software products for the further use in the dynamic identifier to automatically adapt the interface to the identified user.

**Results.** Through the use of information technology in the 2000 unique behavioral portraits were recorded within 30 days, which increased the profitability of the web resource by 20%, ie, demonstrated the effectiveness of the proposed IT.

**Conclusion.** The software provides pseudo-identification of users (building anonymous user base and rules based on their presence in web applications). The main advantages of the software system are the identification model of anonymous users of the final software products for the further use the dynamic identifier to automatically adapt the interface to the identified user.

The main advantage is the lack of open access of similar software aimed at automated adaptation of interfaces based on pseudo-deanitized information about user interfaces.

The prospects for further development include increase the number of identification tokens and the probability of user identification, implementation of the embedded in content management systems and optimization of loss data.

**Keywords:** *interface, adaptive, inductive modelling, web-interface.*

*B.B. Зосімов*, канд. техн. наук, доцент,  
zosimovvv@gmail.com

*А.В. Христордов*, магистр спец. 122 «Компьютерные науки»  
belfegor26@gmail.com



А.С. Булгакова, канд. техн. наук, доцент,  
sashabulgakova2@gmail.com

Кафедра компьютерных наук и прикладной математики Николаевского нац. ун-та  
им. В.А. Сухомлинского, г. Николаев, 54030, ул. Никольская, 24, тел.: (0512) 37-88-09

## ТЕХНОЛОГИЯ АВТОМАТИЗИРОВАННОЙ АДАПТАЦИИ ВЕБ-ПРИЛОЖЕНИЙ НА ОСНОВЕ ИДЕНТИФИКАЦИИ КИБЕРСУЩНОСТЕЙ

**Введение.** Интерфейсы — неотъемлемая составляющая для восприятия и управления программным обеспечением, программными продуктами и связью пользователей с программой. Постоянная интеграция ИТ-технологий в современный быт создает растущий спрос на разработки программных продуктов в области адаптивных интерфейсов, ориентированных на потребности пользователей. Поэтому актуальна задача создания автоматически адаптированного интерфейса под потребности пользователей.

**Цель.** Разработать автоматизированную адаптационную технологию веб-приложений на основе идентификации киберсутностей, где сущность — это набор поведенческих характеристик, собранных на основе запрограммированных на отслеживание изменений объектов интерфейса, персональных настроек и дальнейшее наблюдение за потенциальным изменением настроек интерфейса.

**Методы.** Представлено краткое описание разработки программного обеспечения, направленного на автоматическую адаптацию интерфейсов под потребности пользователей. Программный продукт обеспечивает псевдоидентификацию пользователей (создание анонимной базы пользователей и правил, основанных на их присутствии в веб-приложениях). Основным преимуществом программной системы является модель идентификации анонимных пользователей конечных программных продуктов для дальнейшего использования в динамическом идентификаторе для автоматической адаптации интерфейса к идентифицированному пользователю.

**Результаты.** Благодаря использованию представленной технологии на примере Интернет-магазина, было зафиксировано 2000 уникальных поведенческих портретов в течение 30 дней, что увеличило прибыльность веб-ресурса на 20 процентов.

**Выводы.** Компьютерная программа реализует формирование сущности пользователя из собранной информации об объекте, проявившем активность. Все изменения настроек веб-интерфейса хранятся в базе данных на каждом этапе их осуществления. При использовании идентификатора интерфейсы веб-ресурсов могут динамически изменяться в зависимости от автоматически собранной информации о пользователе.

Перспективы развития предполагают увеличение количества идентификационных токенов для повышения вероятности идентификации пользователей, внедрение встроенных в системы управления контентом и оптимизацию работы с данными.

**Ключевые слова:** интерфейс, адаптивность, индуктивное моделирование, веб-интерфейс.