

## **ПРОГРАМНИЙ ЗАСІБ ЕМУЛЯЦІЇ РОБОТИ ЯДРА ОПЕРАЦІЙНОЇ СИСТЕМИ**

**Смітюх Володимир Тарасович**

магістрант спеціальності «Інформатика» фізико-математичного факультету Тернопільського національного педагогічного університету імені Володимира Гнатюка  
volodyasmitiukh@gmail.com

**Габрусєв Валерій Юрійович**

кандидат педагогічних наук, доцент кафедри інформатики та методики її навчання Тернопільський національний педагогічний університет імені Володимира Гнатюка  
gabrussev@fizmat.tnpu.edu.ua

Успішна експлуатація сучасних комп'ютерних систем практично неможлива без засвоєння основ їх функціонування. Масове використання засобів комп'ютерної техніки, досить часто породжує проблему неоптимального використання наявних як програмних так і апаратних ресурсів, що досить часто призводить до значних матеріальних втрат. Підготовка сучасного фахівця в галузі інформаційних технологій, який вміє побачити названу проблему, оцінити та запропонувати технологію її вирішення неможлива без використання різного роду програмних та апаратних моделей, що дозволяють оцінити поведінку системи в різних умовах, обрати оптимальні параметри її функціонування.

Одним із важливих компонент сучасної комп'ютерної системи є програмна складова, а саме операційна система. ОС можна, розглядати як сукупність компонентів, кожен з яких відповідає за певні функції щодо управління як апаратними так і програмними компонентами системи.

Складність алгоритмічних процесів, реалізованих у ядрі операційної системи, значно ускладнює їх розуміння та подальше прийняття рішень щодо оптимізації використання комп'ютерних систем. Наявність засобів візуальної імітації, емуляції роботи ядра операційної системи значно спрощує вивчення відповідних тем, та дозволяє студентам набути новиків аналізу стану та виробленню рекомендацій щодо оптимального використання ресурсів комп'ютерних систем [1, 3].

Саме тому і виникає проблема у розробці спеціальних програмних засобів навчального призначення для демонстрації роботи алгоритмів роботи окремих компонент ядра операційних систем: планувальник задач, підсистема для роботи з дисковими накопичувачами.

Операційна система (ОС) — комплекс програмних засобів і даних, які забезпечують керування роботою апаратної та програмної складових обчислювальної системи, координують їх взаємодію, забезпечують виконання функції передавання команд і даних між користувачем і комп'ютером [2].

Програми користувача, які створюються для універсальних ЕОМ. обов'язково передбачають можливість передавання управління іншій програмі. Цією програмою є операційна система, точніше її складова, яка забезпечує управління програмами.

Частина програми (у тому числі і ОС), обмежена командами, які дають можливість передавання управління іншій послідовності команд, називається процесом або потоком. Отже, керування процесом виконання програм користувача здійснюється ОС через керування процесами, розподілу між ними ресурсів системи.

Базові компоненти ОС, які відповідають за найважливіші її функції, зазвичай перебувають у пам'яті постійно і виконуються у привілейованому режимі, називають ядром ОС.

Весь код і структури даних ядра перебувають в одному адресному просторі. У ядрі можна виділити кілька функціональних компонентів:

Планувальник процесів – відповідає за реалізацію багатозадачності в системі.

Менеджер пам'яті – виділяє окремий адресний простір для кожного процесу і реалізує підтримку віртуальної пам'яті.

Віртуальна файлова система – надає універсальний інтерфейс взаємодії з різними файловими системами та пристроями введення-виведення.

Драйвери пристроїв – забезпечують безпосередню роботу з периферійними пристроями. Доступ до них здійснюється через інтерфейс віртуальної файлової системи.

Мережний інтерфейс – забезпечує доступ до реалізації мережних протоколів і драйверів мережних пристроїв.

Підсистема міжпроцесової взаємодії – пропонує механізми, які дають змогу різним процесам у системі обмінюватись даними між собою.

Деякі із цих підсистем є логічними компонентами системи, вони завантажуються у пам'ять разом із ядром і залишаються там постійно. Компоненти інших підсистем доцільно реалізувати таким чином, щоб їх код завантажувався у пам'ять за вимогою [1, 2, 3].

Розроблений програмний засіб EmuOs, який імітує роботу планувальника ядра ОС. Даний проект може використовуватись у навчальних цілях для демонстрації та глибшого розуміння студентами роботи багатозадачної ОС з одним процесом і також під час дослідження або моделювання роботи операційної системи з різними вхідними параметрами.

Розглядуваний програмний засіб містить можливості:

імітації роботи ядра ОС;

дослідження роботи ОС з різними параметрами;

дослідження характеристик стратегій.

Важливими особливостями програми EmuOs є:

відсутність необхідності інсталяції;

для коректної роботи не потрібна наявність додаткових програм та бібліотек;

інтерфейс програми достатньо звичний та зручний у використанні;

Основним завданням розробки полягало в створенні програми, яка імітує роботу планувальника ядра ОС та моделювання її роботи в практичних ситуаціях, що дозволяє більш глибоко зрозуміти теоретичні та практичні проблеми які виникають під час проектування операційних систем та їх супроводу.

## Список використаних джерел

1. Таненбаум Э., Вудхалл А. Операционные системы: разработка и реализация. Классика CS. — СПб.: Питер. 2006. 576 с..
2. В. В. Лапінський, В. Ю. Габрусев, Н. Я. Бачинська Основи операційних систем. Посібник для студентів. Тернопіль: Навчальна книга – Богдан, 2002. 80 с.
3. Шеховцов В. А. Операційні системи. — К.: Видавнича група BHV, 2005. 576 с.

## РОЗРОБКА ДОДАТКУ ПІД LARAVEL ДЛЯ ПОШУКУ ДАНИХ В ELASTICSEARCH

### Яніцький Андрій Іванович

магістрант спеціальності «Середня освіта. Інформатика»,  
Тернопільський національний педагогічний університет імені Володимира Гнатюка  
cool.ksandr2013@gmail.com

### Вельгач Андрій Володимирович

доцент кафедри інформатики і методики її викладання,  
Тернопільський національний педагогічний університет імені Володимира Гнатюка  
velgandr@ukr.net

На сьогодні в мережі Інтернет зберігаються величезні об'єми інформації, яку потрібно зберігати, шукати, змінювати, і все це повинно робитись швидше тому що кінцевий користувач зараз зі значно меншою терпимістю ставиться до повільних ресурсів. Сам по собі фреймворк Laravel досить швидким, про те при роботі з великим обсягом даних MySQL не може справитись достатньо швидко, тому потрібна альтернатива

Оскільки Elasticsearch не є надійним сховищем для даних, нам потрібно організувати декілька етапів обробки даних

Дані повинні зберігатись в MySQL

Дані з MySQL повинні бути скопійованими в Elasticsearch

Усі запити від користувачів повинні опрацьовуватись Elasticsearch'ем

Зміна та додавання даних виконується в MySQL базі, а потім за допомогою скрипта, який виконується CRON'ом переносяться в Elasticsearch

Реалізація інтерфейсу взаємодії з Elasticsearch вимагає дуже багато часу, а реалізація створення аналогічних інструментів роботи з даними як у Laravel вимагатиме ще більше часу, тому ми скористаємось готовим інструментом **Elasticquent** - готова бібліотека, яка створює необхідні інтерфейси та засоби, і дозволяє працювати з даними з Elasticsearch в такий самий спосіб, як з даними з MySQL.

### Підключення Elasticquent

Дана бібліотека легко встановлюється за допомогою composer. Для цього потрібно перейти у директорію проекту та виконати команду **composer require elasticquent/elasticquent**

Після того як файли бібліотеки завантажились у Ваш проект, можна перейти до налаштування самого Laravel.