

**Відокремлений структурний підрозділ
«Ніжинський фаховий коледж Національного університету біоресурсів і
природокористування України»**

ВІДДІЛЕННЯ ЕКОНОМІКИ, ЛОГІСТИКИ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Циклова комісія з комп'ютерної інженерії

ДО ЗАХИСТУ ДОПУЩЕНИЙ

Завідувач відділення економіки,
логістики та інформаційних систем

_____ Тетяна РОМАНЕНКО

«_____» _____ 2025 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проєкту фахового молодшого бакалавра

на тему

**«РОЗРОБКА ОНЛАЙН-СИМУЛЯТОРА BIOS SETUP
UTILITY МАТЕРИНСКОЇ ПЛАТИ ASROCK 760GM-HDV»**

ДП.123.211.014.00 ПЗ

Виконав студент IV курсу, групи КН211
спеціальності 123 Комп'ютерна інженерія

_____ Богдан МАРТИНЕНКО

Керівник _____ Анна КАЛІНІЧЕНКО

**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«НІЖИНСЬКИЙ ФАХОВИЙ КОЛЕДЖ НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ»**

Відділення технічно-енергетичних систем та засобів автоматизації

Циклова комісія з комп'ютерної інженерії

Освітньо-професійний ступінь «Фаховий молодший бакалавр»

Спеціальність 123 Комп'ютерна інженерія

ЗАТВЕРДЖУЮ

Голова циклової комісії

_____ Анна КАЛІНІЧЕНКО

« 24 » лютого 2025 року

**З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ СТУДЕНТУ**

Мартиненку Богдану Вячеславовичу

1. Тема проєкту «Розробка онлайн-симулятора BIOS Setup Utility материнської плати ASRock 760GM-HDV»

керівник проєкту (роботи) Калініченко Анна Олександрівна,

затверджені наказом від «10» лютого 2025 року №12«С»

2. Строк подання студентом проєкту 24 травня 2025 року

3. Вихідні дані до проєкту Технічна документація та специфікації BIOS материнської плати ASRock 760GM-HDV. Структура меню BIOS, параметри конфігурації (CPU, пам'ять, вентилятори, завантаження тощо)

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) _____

1. Організаційна частина.

2. Технологічна частина.

3. Спеціальна частина.

4. Економічна частина.

5. Охорона праці.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Діаграма компонентів Схема електрична структурна

Діаграма послідовностей Схема електрична структурна

Діаграма прецедентів Схема електрична структурна

Структура сайту Схема електрична структурна

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Дата	Підпис
		завдання видав	завдання прийняв студент
1	Калініченко А.О.	24.02.2025	
2	Калініченко А.О.	24.02.2025	
3	Калініченко А.О.	24.02.2025	
4	Калініченко А.О.	24.02.2025	
5	Калініченко А.О.	24.02.2025	

7. Дата видачі завдання 24.02.2025

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Організаційна частина	24.02-13.03	
2	Технологічна частина	14.03-03.04	
3	Спеціальна частина	04.04-24.04	
4	Охорона праці	25.04-08.05	
5	Економічна частина	09.05-24.05	

Студент

(підпис)

Богдан МАРТИНЕНКО

Керівник проєкту

(підпис)

Анна КАЛІНІЧЕНКО

АНОТАЦІЯ

Даний дипломний проєкт спрямований на створення онлайн-симулятора BIOS Setup Utility для материнської плати ASRock 760GM-HDV. Основна мета проєкту надання студентам можливості безпечно ознайомитися з функціоналом BIOS у віртуальному середовищі без ризику змін у реальній системі.

Симулятор точно відтворює інтерфейс і основні функції BIOS, дозволяючи змінювати системний час, частоту процесора, швидкість вентилятора та інші параметри. Керування здійснюється виключно клавіатурою, а збереження налаштувань можливе через LocalStorage.

Також реалізована система завдань, за виконання яких користувач отримує бали, що сприяє кращому засвоєнню матеріалу. Проєкт орієнтований на заклади освіти та компанії, що займаються виробництвом або продажем комп'ютерних комплектуючих.

ANNOTATION

This graduation project is aimed at creating an online BIOS Setup Utility simulator for the ASRock 760GM-HDV motherboard. The main goal of the project is to provide students with the opportunity to safely familiarize themselves with the BIOS functionality in a virtual environment without the risk of changes to the real system.

The simulator accurately reproduces the BIOS interface and basic functions, allowing you to change the system time, processor frequency, fan speed, and other parameters. Controls are carried out exclusively via the keyboard, and settings can be saved via LocalStorage.

A system of tasks is also implemented, for which the user receives points, which contributes to better learning. The project is aimed at educational institutions and companies engaged in the production or sale of computer components.

ЗМІСТ

ВСТУП.....	5
1 ОРГАНІЗАЦІЙНА ЧАСТИНА	7
1.1 Етапи розробки проєкту	7
1.2 Інструменти та технології, використані у проєкті.....	11
1.3 Методи організації та підхід до виконання проєкту	14
2 ТЕХНОЛОГІЧНА ЧАСТИНА	32
2.1 Вибір технологій для реалізації симулятора BIOS.....	18
2.2 Аналіз існуючих рішень	19
2.3 Аналоги BIOS	22
2.4 Використання LocalStorage для збереження налаштувань	25
2.5 Система завдань у симуляторі BIOS	26
2.6 Вибір редактора коду.....	27
3 СПЕЦІАЛЬНА ЧАСТИНА.....	32
3.1 Загальний вид програми.....	32
3.2 Можливість збереження та завантаження профілів користувачів.....	38
3.3 Система паролів для доступу до налаштувань	41
3.4 Система завдань для покращення навичок роботи з BIOS.....	44
3.5 Навігація та управління	47
3.6 Дрібні функціональні можливості.....	50
4 ЕКОНОМІЧНА ЧАСТИНА.....	54
5 ОХОРОНА ПРАЦІ	58
ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	66

					ДП.123.211.014.00 ПЗ			
<i>Зм.</i>	<i>Арк.</i>	<i>№ документа</i>	<i>Підпис</i>	<i>Дата</i>	Розробка онлайн-симулятора BIOS Setup Utility материнської плати ASRock 760GM-HDV Пояснювальна записка	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
<i>Розроб.</i>		<i>Мартиченко Б.В.</i>					4	67
<i>Перевір.</i>		<i>Калініченко А.О.</i>						
<i>Реценз.</i>								
<i>Н. Контр.</i>		<i>Кочур Д.О.</i>						
<i>Затверд.</i>		<i>Калініченко А.О.</i>				<i>НФК гр.КН211</i>		

ВСТУП

Актуальність теми дипломного проєкту зумовлена недостатньою обізнаністю багатьох студентів щодо важливості та функціональності BIOS. BIOS (Basic Input/Output System) є основним інтерфейсом для налаштування параметрів комп'ютерної системи до завантаження операційної системи, однак не кожен користувач має достатньо знань для його ефективного використання. Часто користувачі BIOS використовують лише за необхідності, не знаючи про весь спектр можливостей, які він надає для налаштування продуктивності, енергозбереження, безпеки та інших важливих параметрів. Враховуючи це, проєкт, присвячений розробці онлайн-симулятора BIOS Setup Utility для материнської плати ASRock 760GM-HDV, є надзвичайно актуальним, оскільки він дозволяє студентам отримати необхідні знання та навички для безпечного і ефективного користування BIOS в онлайн-середовищі без страху зробити помилку, яка може вплинути на реальні налаштування комп'ютера.

Метою даного дипломного проєкту є розробка онлайн-симулятора, який дозволяє користувачам вивчати та практикувати роботу з основними функціями BIOS, такими як налаштування частоти процесора, швидкості вентилятора, параметрів пам'яті, часу і дати системи та інших. Важливою особливістю є створення симулятора, який дає можливість навчатися без ризику змінити налаштування реального комп'ютера, що робить процес навчання безпечним і доступним. За допомогою цього інструменту студенти можуть на практиці освоїти основи роботи з BIOS, що значно спростить їм подальше навчання та роботу з реальними комп'ютерними системами.

Об'єктом дослідження є функціонування BIOS, його роль у налаштуванні комп'ютерної системи, а предметом – створення освітнього інструменту, який дозволяє вивчати та освоювати роботу з BIOS у безпечному середовищі. Проєкт надає можливість практичного застосування через симуляцію реальних завдань.

					ДП.123.211.014.00 ПЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

Для досягнення поставленої мети були використані методи програмування, управління профілями користувачів через функції saveProfile та loadProfile, що дозволяють зберігати та відновлювати налаштування за допомогою LocalStorage і JSON. Також було розроблено механізм роботи з інтерфейсом користувача через маніпуляції з DOM (Document Object Model), що забезпечує взаємодію з елементами сайту. Для зручності користувача реалізовано систему анімацій та логування, що покращує взаємодію та спрощує розуміння виконаних дій. Окрім того, проєкт включає функціонал для створення та завантаження профілів користувачів, а також для виконання завдань, що дозволяють отримувати бали за виконання конкретних дій у BIOS, що мотивує користувачів до глибшого вивчення та розуміння його роботи.

Структура роботи складається з кількох розділів, які детально описують процес розробки проєкту, технічні особливості реалізації симулятора, а також способи взаємодії з користувачем через інтерфейс. Однією з ключових частин є система завдань, яка дозволяє користувачам виконувати конкретні дії в BIOS та отримувати за це бали, що сприяє кращому засвоєнню функцій BIOS і розвитку практичних навичок.

Таким чином, результатом даної роботи став освітній інструмент, який дозволяє безпечним і зручним способом освоїти основи роботи з BIOS, що є важливим для подальшого навчання студентів. Це допоможе їм не тільки зберегти час на вивчення складних технологій, але й отримати корисні навички, які можуть бути корисні для майбутньої кар'єри в галузі інформаційних технологій.

					ДП.123.211.014.00 ПЗ	Арк.
						6
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

1 ОРГАНІЗАЦІЙНА ЧАСТИНА

1.1 Етапи розробки проєкту

Розробка онлайн-симулятора BIOS Setup Utility для материнської плати ASRock 760GM-HDV є важливим етапом в освітньому процесі, оскільки дозволяє студентам та учням краще зрозуміти та освоїти функціональність BIOS. Проєкт був розроблений для того, щоб допомогти користувачам освоїти основні функції BIOS без ризику змінити реальні налаштування системи. Першим кроком у розробці стала ретельна підготовка до реалізації проєкту, що включала планування, вибір методів та інструментів для забезпечення максимальної ефективності та зручності користувача.

Для цього проєкту було використано методи розробки, які забезпечують зручність у взаємодії з програмою і дозволяють реалізувати функціональність, схожу на реальний BIOS. Одним із основних аспектів стало відтворення інтерфейсу, який максимально нагадує оригінальну утиліту BIOS. Це дозволяє користувачам почуватися комфортно, адже вони мають справу з інтерфейсом, знайомим з реального обладнання. Для цього проєкту було використано сучасні технології веб-розробки, які дозволили створити інтерфейс, що відтворює функціональність BIOS на веб-сторінці.

Першочерговим завданням було визначення основних функціональних блоків проєкту, таких як управління параметрами BIOS, збереження та завантаження налаштувань через LocalStorage, а також створення системи завдань, яка дозволяє користувачам отримувати бали за виконання певних дій. Одним із ключових моментів стало забезпечення збереження користувацьких налаштувань, що дозволяє користувачам зберігати свою конфігурацію BIOS без страху втратити або змінити важливі параметри.

Наступним етапом було проектування структури меню, яка б максимально відповідала реальному інтерфейсу BIOS. Це дозволило створити організовану

					ДП.123.211.014.00 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

навігацію, де користувач може переміщатися між розділами, вибирати параметри та змінювати їх. Для цього було використано прості і зручні методи навігації через клавіатуру, що є основним способом взаємодії з реальним BIOS. Цей підхід дозволив максимально наблизити симулятор до реального використання BIOS, що є важливим для освітнього процесу.

Завдання, які були включені до проєкту, допомогли не лише практично застосувати знання про BIOS, а й дозволили користувачам отримувати бали за виконання кожного завдання. Такі завдання, як «Встановіть час у BIOS на 20:53:42», дозволяють користувачам здобувати знання у процесі практичних дій. Оскільки завдання є частиною освітнього процесу, вони сприяють закріпленню матеріалу та розвитку практичних навичок роботи з BIOS.

Проєкт було спроектовано таким чином, щоб максимально зберегти реалістичність BIOS, але при цьому зробити взаємодію з програмою доступною та безпечною для користувачів. Завдяки цьому користувач може змінювати параметри, не турбуючись про наслідки своїх дій, оскільки все це відбувається в безпечному онлайн-середовищі. Для збереження налаштувань користувач може скористатися функціями збереження профілю, що дозволяє йому відновити раніше збережені налаштування в будь-який час.

Також, для поліпшення взаємодії з користувачем, у проєкті було використано бібліотеку Toast.js, що дозволяє виводити повідомлення про дії користувача. Це робить використання симулятора ще більш зручним і інтуїтивно зрозумілим. Завдяки цій бібліотеці користувачі отримують повідомлення про успішне збереження налаштувань, виконання завдання та інші важливі дії, що робить роботу з симулятором більш зрозумілою та приємною.

Протягом розробки проєкту велика увага була приділена тестуванню всіх функцій, щоб переконатися у правильності роботи кожного елемента та забезпечити безпеку взаємодії з користувачем. Це включало перевірку кожного етапу роботи симулятора, від зміни налаштувань BIOS до виконання завдань і накопичення балів. Зокрема, тестування охоплювало всі аспекти, пов'язані з

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

роботою з LocalStorage, а також перевірку відображення повідомлень за допомогою бібліотеки Toast.js.

Для цього проєкту було використано кілька ключових методів, які забезпечили ефективну реалізацію всіх функцій. Серед них можна виокремити використання технологій для роботи з DOM, що дозволяє динамічно змінювати контент на сторінці, а також методи обробки подій, що дозволяють користувачеві взаємодіяти з симулятором. Також важливою складовою частиною є використання JavaScript для збереження та завантаження профілів через LocalStorage, що дозволяє зберігати налаштування між сесіями.

Проєкт був створений з урахуванням всіх вимог і стандартів, що забезпечує його ефективність та зручність у використанні (рис. 1.1). Він надає користувачам можливість безпечно вивчати основи роботи з BIOS, надаючи їм повну свободу змінювати параметри та виконувати завдання, не побоюючись змінити налаштування реального комп'ютера (рис. 1.2).

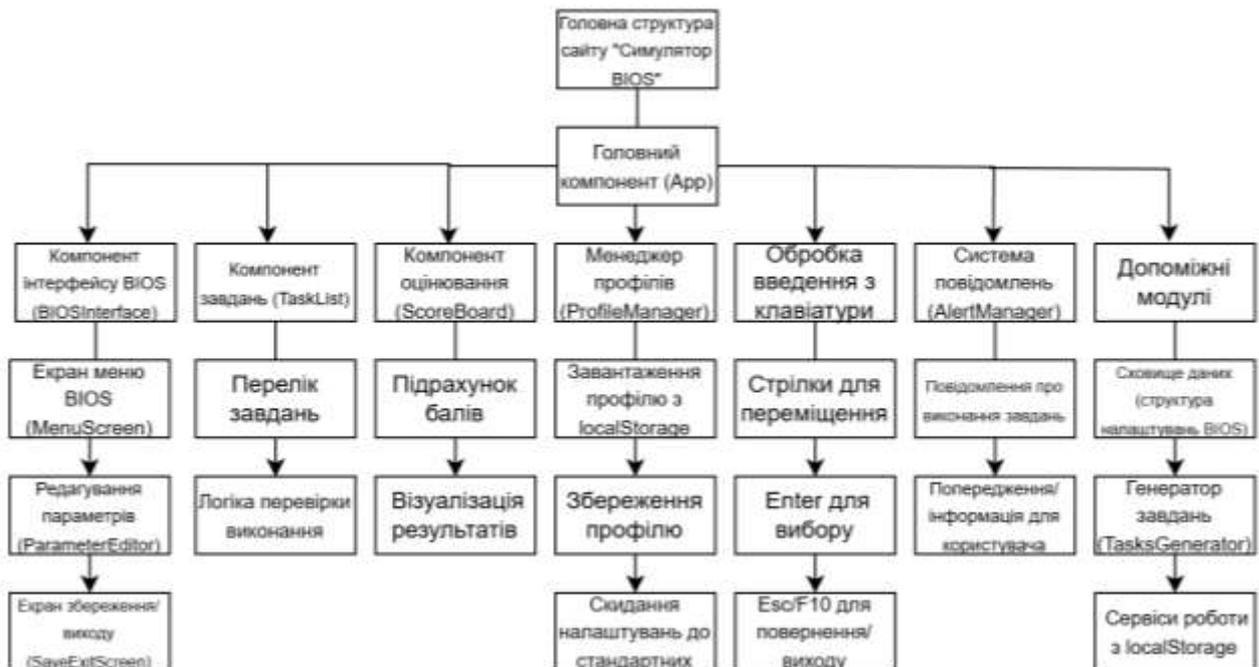


Рисунок 1.1 – Структура сайту

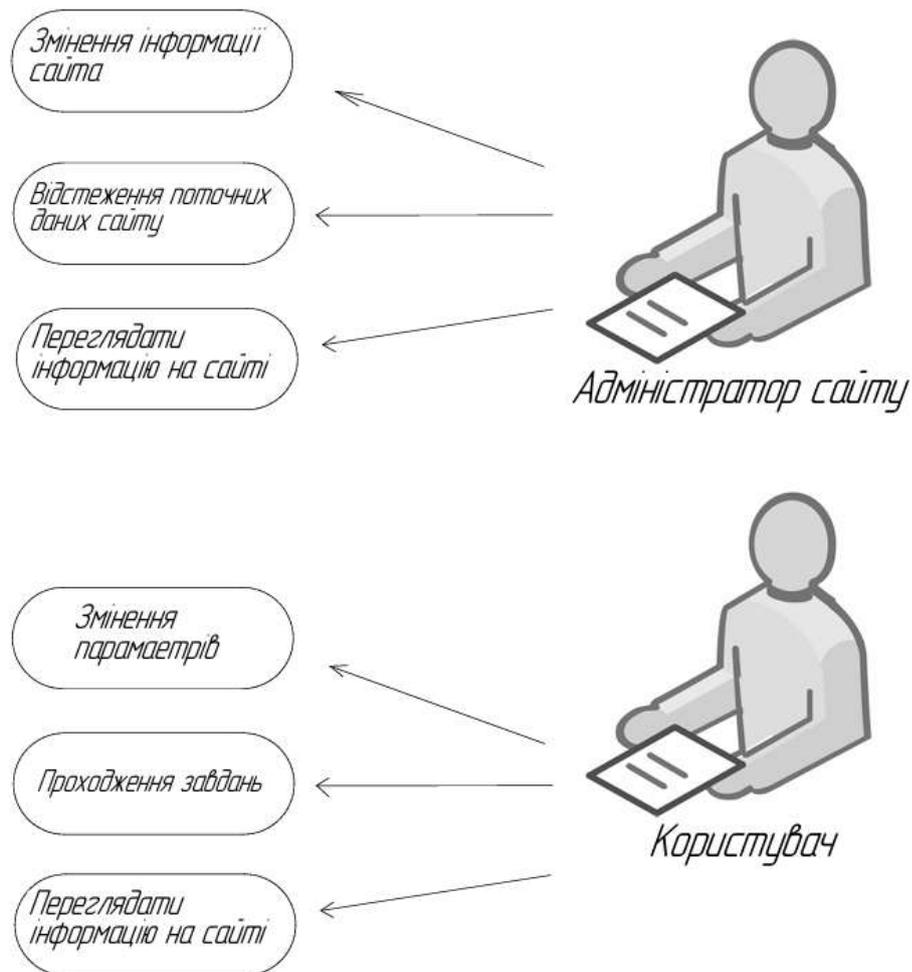


Рисунок 1.2 – Діаграма прецедентів

Для кращого розуміння роботи системи створено діаграму послідовностей (рис. 1.3), яка ілюструє процеси збереження (Save 1st User Defaults) та завантаження (Load 1st User Defaults) профілю користувача. Ця діаграма наочно демонструє алгоритм взаємодії між користувачем, інтерфейсом, логікою програми та сховищем даних, показуючи, як саме відбувається зчитування та збереження параметрів BIOS.

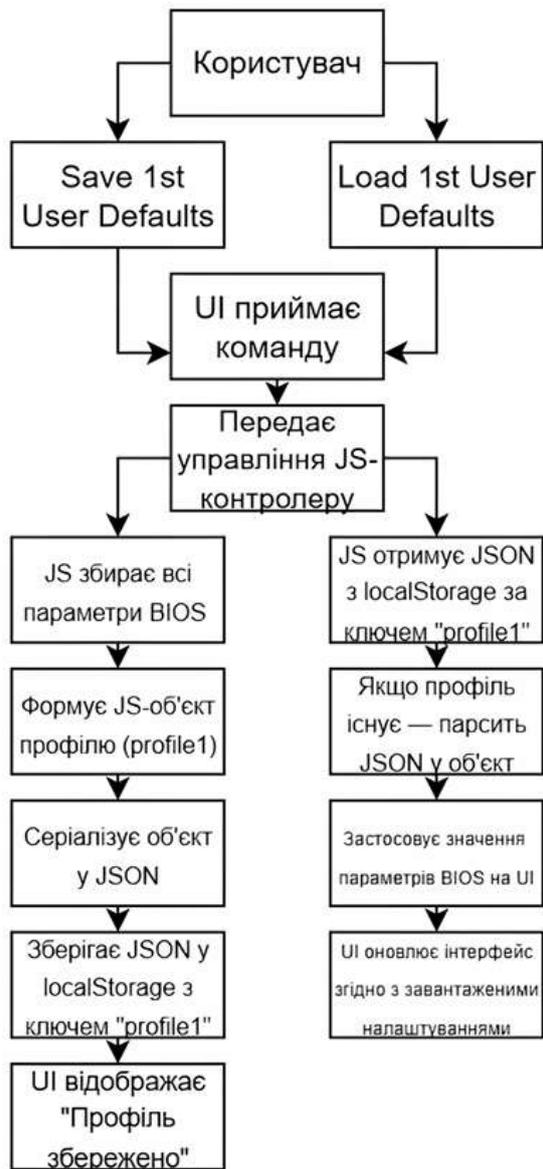


Рисунок 1.3 – Діаграма послідовностей

1.2 Інструменти та технології, використані у проєкті

Розробка онлайн-симулятора BIOS Setup Utility для материнської плати ASRock 760GM-HDV (рис. 1.4) вимагала ретельного планування, оскільки проєкт передбачав відтворення функціональності реального BIOS в онлайн-середовищі. Для досягнення цієї мети необхідно було вибрати оптимальні методи та інструменти, щоб забезпечити максимальну точність відтворення інтерфейсу та функцій BIOS, а також зручність використання для користувачів. Кожен етап

проєкту був продуманий так, щоб максимально спростити взаємодію користувача з програмою та забезпечити йому можливість навчатися і виконувати завдання без ризику змінити налаштування реальної системи.

Одним із важливих аспектів було створення структури проєкту, що дозволило б зберігати гнучкість у його розширенні та оновленні.

Проєкт був організований таким чином, щоб усі функції BIOS можна було швидко і легко додавати або змінювати, не порушуючи загальної логіки роботи програми. Для цього було використано методи роботи з DOM (Document Object Model), які дозволяють динамічно змінювати контент на веб-сторінці та створювати інтерфейс, схожий на реальний BIOS. Зокрема, це стосувалося створення меню для зміни налаштувань, відображення параметрів, таких як частота процесора, час і дата, швидкість вентилятора, а також системи завдань, де користувачі отримують бали за виконання певних дій.

Проєкт передбачав створення функціоналу для збереження і завантаження налаштувань, щоб користувачі могли працювати з власними профілями. Для цього було реалізовано використання LocalStorage, що дозволяє зберігати налаштування між сесіями без необхідності постійно вводити їх вручну. Це стало важливим етапом проєкту, оскільки збереження налаштувань дає користувачам можливість відновити свої попередні конфігурації та продовжити роботу з того місця, де вони зупинилися. Крім того, для реалізації функції збереження та завантаження налаштувань було використано формати JSON, що дозволило створити зручний і гнучкий механізм для роботи з даними.

Невід'ємною частиною проєкту стала система завдань, яка дозволяє користувачам отримувати бали за виконання певних дій в BIOS. Завдання були спроектовані так, щоб вони не лише допомагали користувачам вивчати основи налаштування BIOS, а й давали можливість перевірити набуті знання на практиці. Одним із таких завдань стало встановлення точного часу в BIOS, що дозволяє користувачам отримати бали за точність виконання завдання. Усього в проєкті

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

передбачено до 100 балів, що стимулює користувачів виконувати завдання і вдосконалювати свої навички роботи з BIOS.

При розробці інтерфейсу було важливо забезпечити максимальну схожість з реальним BIOS, щоб користувачі могли без труднощів переходити від одного параметра до іншого, змінюючи налаштування зручним способом. Тому інтерфейс був спрощений, щоб користувач міг зосередитися на основних функціях BIOS і не відволікатися на зайві елементи.

Оскільки в реальному BIOS основне управління здійснюється за допомогою клавіатури, було вирішено застосувати аналогічний підхід і в симуляторі. Таким чином, для навігації між параметрами BIOS користувачам достатньо використовувати клавіші вліво, вправо, вгору і вниз, що робить взаємодію з симулятором інтуїтивно зрозумілою.

Важливою частиною проєкту стало також забезпечення надійності та стабільності роботи симулятора. Оскільки користувачі мали змогу змінювати параметри BIOS, зокрема такі важливі як частота процесора чи швидкість вентилятора, потрібно було створити такі механізми, що дозволяють унеможливити неправильні або небезпечні зміни. Для цього було реалізовано ряд перевірок і обмежень, які гарантують, що зміни в параметрах BIOS не призведуть до помилок або збоїв у роботі симулятора.

Також, незважаючи на те, що проєкт був орієнтований на освітні цілі, важливо було створити такий інтерфейс, який би відповідав усім вимогам користувачів щодо зручності та функціональності. Для цього було приділено велику увагу простоті використання, щоб навіть користувачі без досвіду роботи з BIOS могли швидко зрозуміти, як користуватися програмою. Це забезпечило успішне залучення до проєкту студентів, які могли без проблем освоїти основи роботи з реальним BIOS.

Важливим аспектом розробки проєкту також стало тестування кожного з його елементів, щоб переконатися в їхній коректності та ефективності. Протягом тестування було перевірено всі функції, зокрема зміну налаштувань, виконання

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

завдань і збереження профілів. Це дозволило виправити можливі помилки та вдосконалити функціональність симулятора, забезпечуючи його стабільну роботу під час використання. Крім того, тести допомогли виявити можливі проблеми, пов'язані з відображенням даних або використанням бібліотеки Toast.js, що позитивно позначилося на загальній якості роботи проєкту.

1.3 Методи організації та підхід до виконання проєкту

Аналіз предметної області є важливою частиною підготовки до розробки будь-якого програмного продукту, особливо коли мова йде про симулятор, який має на меті освітні цілі. Для проєкту онлайн-симулятора BIOS Setup Utility для материнської плати ASRock 760GM-HDV (рис. 1.4) було необхідно детально вивчити, як працює реальний BIOS, які його основні функції та як їх можна відтворити в онлайн-форматі. BIOS (Basic Input/Output System) є низькорівневою системою, що дозволяє взаємодіяти з апаратним забезпеченням комп'ютера, і його налаштування відіграють важливу роль у функціонуванні системи.



Рисунок 1.4 – Материнська плата ASRock 760GM-HDV

Одним із ключових моментів у розробці цього проєкту стало розуміння того, як користувачі взаємодіють з BIOS у реальному житті. У більшості випадків,

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

користувачі, зокрема студенти, не мають досвіду роботи з BIOS, оскільки ці налаштування рідко змінюються після першої установки операційної системи. Однак для навчання важливо не лише вивчити базові функції BIOS, а й розуміти, як правильно налаштовувати системні параметри для досягнення оптимальної продуктивності або енергозбереження. Такий підхід дозволяє студентам краще підготуватися до реальних завдань у сфері комп'ютерних технологій, де знання та навички роботи з BIOS можуть стати важливими.

Щоб реалізувати цю мету, було розглянуто основні функції BIOS, які повинні бути представлені в онлайн-симуляторі. Найважливішими з них є налаштування часу й дати, налаштування частоти процесора, керування швидкістю вентилятора, параметри пам'яті, а також налаштування безпеки, такі як створення паролів для доступу до BIOS. Ці функції дозволяють користувачеві змінювати базові параметри системи, що є важливим для розуміння того, як вони впливають на роботу комп'ютера. Для правильної реалізації цих функцій важливо було забезпечити їхню коректну інтерпретацію в онлайн-форматі, щоб користувач міг навчитися змінювати налаштування без ризику для системи.

Аналіз предметної області також включав вивчення існуючих онлайн-симуляторів та навчальних інструментів, які мають схожу мету. Багато з цих інструментів надають базову інформацію про BIOS, але не дозволяють виконувати практичні завдання, які сприяють глибшому розумінню роботи BIOS. Враховуючи це, було вирішено впровадити систему завдань, де користувачі можуть отримувати бали за виконання конкретних налаштувань. Це не лише сприяє закріпленню знань, а й дозволяє створити мотивацію для виконання завдань, що допомагає користувачам краще засвоїти матеріал. Завдання включають як прості операції, такі як встановлення часу, так і складніші, пов'язані з налаштуванням продуктивності чи енергозбереження.

У рамках аналізу предметної області було також розглянуто особливості реалізації інтерфейсу симулятора, який повинен бути максимально схожим на

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

реальний BIOS. Для реалізації цього проєкту було обрано материнську плату ASRock 760GM-HDV, оскільки вона є оптимальним варіантом для створення онлайн-симулятора BIOS. Головним критерієм вибору стало її поширення серед користувачів, а також доступність документації та можливість точно відтворити функціональність BIOS у симуляторі.

Материнська плата ASRock 760GM-HDV використовує BIOS із класичним текстовим інтерфейсом, що робить її зручною для навчання студентів. Вона містить стандартний набір налаштувань, які є у більшості BIOS, що дозволяє використовувати симулятор як універсальний навчальний інструмент.

Також враховувався баланс між функціоналом і простотою роботи. ASRock 760GM-HDV має всі основні параметри, необхідні для освітнього процесу: налаштування процесора, оперативної пам'яті, швидкості обертання вентиляторів, параметри безпеки та завантаження системи. У той же час її інтерфейс не перевантажений зайвими функціями, що могло б ускладнити процес навчання.

Були вивчені специфікації реальних BIOS, а також досвід користувачів, щоб зрозуміти, які функції є найважливішими для навчання. Важливо було забезпечити користувачеві інтуїтивно зрозумілий інтерфейс, де кожна дія матиме чітке пояснення і не залишатиме місця для помилок. Враховуючи, що основне керування в BIOS здійснюється за допомогою клавіатури, було вирішено використовувати подібний підхід і в симуляторі, що дозволяє користувачу зручніше взаємодіяти з програмою.

Також важливим аспектом аналізу предметної області стало дослідження безпеки та стабільності роботи симулятора. Під час виконання завдань користувач може змінювати налаштування, які в реальному BIOS можуть вплинути на стабільність системи. Однак, оскільки цей проєкт є лише симулятором, то завдання полягало в тому, щоб створити безпечне середовище, де користувач може змінювати налаштування, не побоюючись пошкодити систему. Це

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

досягалося через використання обмежень на зміни параметрів, які могли б спричинити несправності в реальному середовищі.

Окрім цього, аналіз предметної області включав вивчення можливостей для подальшого розвитку та розширення функціоналу симулятора. Адже, хоча на початковому етапі проекту було передбачено лише базові функції, у майбутньому проєкт може бути доповнений новими завданнями, параметрами та іншими можливостями. Враховуючи швидкий розвиток технологій і зміну потреб користувачів, важливо було створити систему, що дозволяє легко додавати нові функції без великих змін в основному коді.

Загалом, аналіз предметної області допоміг чітко визначити основні цілі та завдання проєкту, а також зрозуміти, які функції та можливості повинні бути реалізовані в симуляторі для досягнення найкращих результатів у навчанні.

					ДП.123.211.014.00 ПЗ	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		17

2 ТЕХНОЛОГІЧНА ЧАСТИНА

2.1 Вибір технологій для реалізації симулятора BIOS

Розробка онлайн-симулятора BIOS Setup Utility потребувала ретельного підбору технологій, які забезпечать ефективність роботи, швидкодію та коректну взаємодію з користувачем. Для реалізації було обрано HTML та CSS JavaScript, що дозволило створити повноцінний веб-додаток, доступний у браузері без потреби в додатковому програмному забезпеченні. Основний функціонал симулятора реалізовано засобами нативного JavaScript, що забезпечує максимальну гнучкість у розробці.

Головна особливість проекту – можливість збереження налаштувань BIOS у локальному сховищі браузера. Для цього використано LocalStorage, який дозволяє записувати та зчитувати дані без підключення до серверів. Збереження параметрів реалізовано у вигляді JSON-об'єктів, що дозволяє легко їх обробляти та змінювати. Основні функції збереження (saveProfile()) та завантаження (loadProfile()) виконують роботу із LocalStorage, зберігаючи такі параметри, як встановлений час, дата, частота процесора та швидкість вентилятора.

Ще однією важливою особливістю є система керування симулятором, яка здійснюється виключно за допомогою клавіатури, що відповідає реальному BIOS. Основні клавіші дозволяють переміщуватися між пунктами меню, змінювати значення параметрів та підтверджувати або скасовувати зміни. Використання лише клавіатурного введення робить симулятор максимально реалістичним і зручним для користувачів, які готуються працювати з реальним BIOS.

Інтерфейс симулятора повністю відтворює класичний вигляд BIOS. Використано чітку, контрастну колірну схему, характерну для текстових меню BIOS, що створює автентичне візуальне сприйняття. Дизайн реалізований засобами CSS без використання сторонніх бібліотек, що мінімізує навантаження на браузер і підвищує швидкість роботи симулятора.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

Для полегшення навчання реалізовано систему завдань, які користувач може виконувати, змінюючи параметри BIOS. Наприклад, завдання може вимагати встановити певний час або змінити швидкість вентилятора. Після виконання завдання система автоматично перевіряє правильність введених значень та нараховує відповідну кількість балів. Максимально можна набрати 100 балів.

Для покращення взаємодії з користувачем у проєкті використано бібліотеку Toast.js, яка забезпечує зручні сповіщення про виконані дії. Вона дозволяє швидко повідомляти користувача про успішне виконання завдання, зміну параметрів або збереження налаштувань. Завдяки цьому освітній процес стає більш інтерактивним і зручним.

Реалізація симулятора BIOS стала можливою завдяки ефективному використанню сучасних веб-технологій. Використання нативного JavaScript дозволило досягти високої продуктивності та сумісності з більшістю браузерів, а застосування LocalStorage зробило збереження параметрів швидким і зручним. Симулятор BIOS забезпечує користувачів можливістю вивчати та практикуватися в керуванні BIOS у безпечному середовищі, що є важливою складовою освітнього процесу.

2.2 Аналіз існуючих рішень

На сучасному ринку існує обмежена кількість веб-симуляторів BIOS, і більшість із них є застарілими як з точки зору інтерфейсу, так і функціональності. Під час аналізу існуючих рішень було виявлено кілька онлайн-версій BIOS, проте всі вони мали суттєві недоліки.

Один із знайдених варіантів пропонував графічний інтерфейс, керований мишею, що не відповідає класичному управлінню BIOS через клавіатуру. Крім того, у ньому була застаріла версія BIOS, яка не містила багатьох важливих налаштувань, характерних для сучасних материнських плат. Інший аналог вже

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

підтримував управління за допомогою клавіатури, що є більш коректним підходом, однак він також мав застарілий дизайн і сильно урізаний функціонал.

Головні проблеми наявних рішень:

1) Застарілий інтерфейс – більшість доступних симуляторів BIOS відтворюють версії 1990–2000-х років, які не відповідають сучасним стандартам.

2) Неповний функціонал – відсутність важливих розділів налаштувань, таких як конфігурація чіпсета, управління вентиляторами, безпека та інші параметри, які є у сучасних версіях BIOS.

3) Відсутність збереження параметрів – усі зміни скидаються при оновленні сторінки, що робить такі симулятори непридатними для навчання або тестування.

4) Незручність у використанні – деякі аналоги працюють лише з мишею, що не відповідає реальному використанню BIOS.

Функціональність онлайн-симулятора BIOS базується на технологіях, які забезпечують зручність користування, ефективну обробку даних та реалістичне відтворення роботи BIOS. Основною мовою програмування обрано JavaScript, що дозволяє реалізувати всі необхідні механізми керування та взаємодії користувача з інтерфейсом. HTML використовується для побудови структури сторінки, а CSS відповідає за стилізацію та відтворення візуального оформлення, максимально наближеного до оригінального BIOS.

Управління симулятором здійснюється виключно за допомогою клавіатури, що повністю відповідає реальному BIOS. Реалізація цієї функції

базується на обробці подій клавіатури через JavaScript (keydown, keyup). Відповідні функції перевіряють натискання клавіш, реагують на зміни положення курсора, вибір пунктів меню та зміну значень параметрів. Наприклад, клавіші зі стрілками (ArrowUp, ArrowDown, ArrowLeft, ArrowRight) переміщують курсор між елементами, клавіша Enter підтверджує вибір, а Esc дозволяє виходити із розділів без збереження змін.

					ДП.123.211.014.00 ПЗ	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		20

Для збереження змін у параметрах BIOS використано LocalStorage, який дозволяє записувати дані в локальну пам'ять браузера без використання серверних технологій. Доступ до даних здійснюється через методи `setItem()` та `getItem()`, що дозволяє легко зчитувати і зберігати інформацію про користувацькі налаштування. У вигляді JSON-об'єкта зберігаються такі параметри, як встановлений час, дата, швидкість вентилятора, частота процесора та інші критично важливі налаштування, які можна редагувати у межах симулятора.

Крім стандартних можливостей BIOS, у симуляторі реалізована система завдань, що дозволяє користувачам отримувати практичний досвід. Завдання формуються у вигляді текстових інструкцій, які вимагають виконання певних змін у налаштуваннях. Наприклад, користувач може отримати завдання встановити певний час або змінити параметри живлення. Після виконання завдання система автоматично перевіряє правильність виконаних дій і нараховує відповідні бали.

Щоб забезпечити зручність користування, у симуляторі реалізована система повідомлень, яка допомагає користувачеві орієнтуватися в інтерфейсі. Для цього використано бібліотеку `Toast.js`, яка дозволяє створювати спливаючі повідомлення про дії користувача. Наприклад, після успішного збереження параметрів з'являється повідомлення про те, що налаштування збережено, а при неправильному виконанні завдання відображається відповідний інформативний меседж.

Графічне відображення BIOS виконано за допомогою CSS-стилізації, що дозволяє імітувати характерний вигляд інтерфейсу. Використано спеціальні кольорові схеми, такі як синій фон і білий або жовтий текст, що створює ефект роботи з реальним BIOS. Також реалізовано затемнення неактивних елементів та підсвічування активних пунктів меню, що допомагає користувачеві швидко орієнтуватися в інтерфейсі.

Для забезпечення плавної взаємодії із симулятором використано методи керування відображенням елементів, зокрема `style.display` та `style.opacity`. Це дозволяє приховувати або відображати окремі блоки інтерфейсу, забезпечуючи

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

логіку переходів між меню та підменю. Анімаційні ефекти додано для більш плавного відтворення змін, що покращує користувацький досвід.

2.3 Аналоги BIOS

В мережі можна знайти декілька аналогів онлайн-симуляторів BIOS, які частково відтворюють функціональність базової системи введення-виведення. Проте більшість з них мають значні обмеження у функціоналі, застарілий інтерфейс, низьку зручність використання та відсутність важливих можливостей, необхідних для повноцінного навчального процесу. Це значно знижує їх ефективність як навчальних інструментів.

Під час аналізу існуючих рішень було знайдено два основні аналоги. Перший з них мав графічний інтерфейс, який дозволяв керувати BIOS за допомогою миші, що є неприйнятним для класичних версій BIOS (рис. 2.1). У більшості реальних систем взаємодія з BIOS здійснюється виключно за допомогою клавіатури, тому подібна реалізація суперечить справжньому принципу роботи BIOS. Крім того, навігація в такому симуляторі була досить незручною, що ускладнювало його використання. Окремі елементи меню працювали некоректно, а деякі важливі параметри були відсутніми, що обмежувало можливості користувачів.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

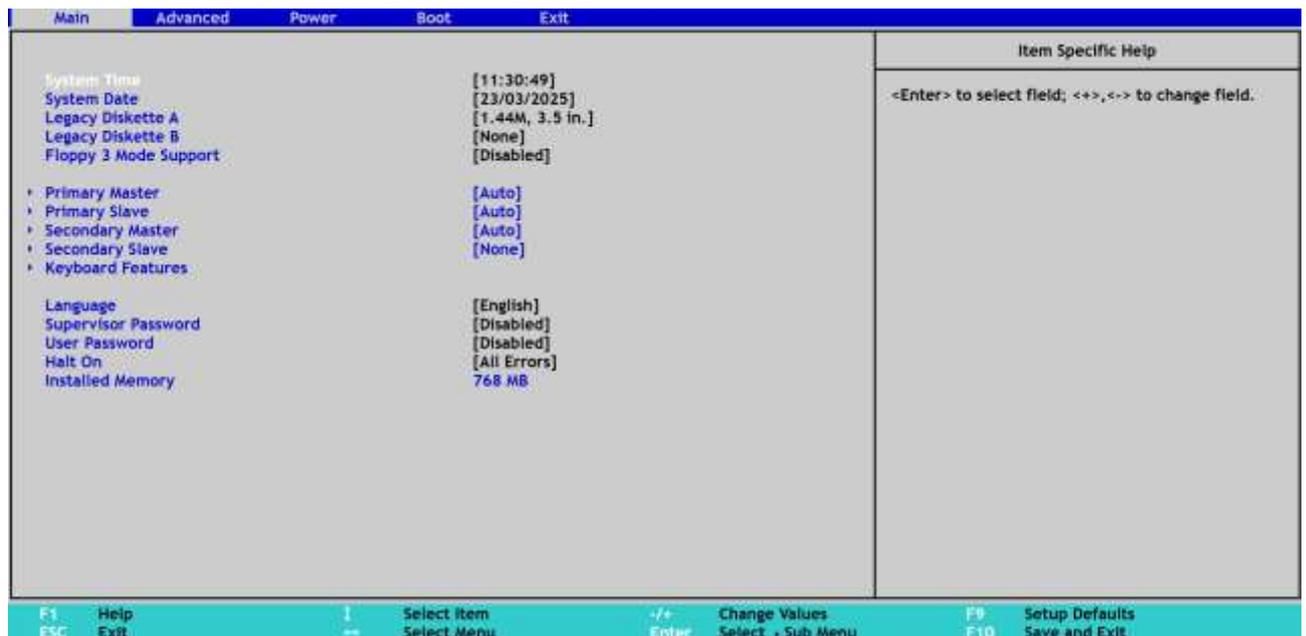


Рисунок 2.1 – Анлог BIOS №1

Другий аналог виявився більш наближеним до реального BIOS, оскільки підтримував управління клавіатурою, що відповідає традиційному способу взаємодії з BIOS. Однак і цей варіант мав серйозні недоліки. Одним із головних мінусів була застаріла версія BIOS, яка нагадувала інтерфейси кінця 1990-х — початку 2000-х років (рис. 2.2).

Дизайн такого симулятора виглядав застарілим, а функціонал був сильно обмежений. Деякі важливі налаштування, що є ключовими в сучасних BIOS, просто були відсутні, що знижувало навчальну цінність такого симулятора.

Головними недоліками знайдених аналогів є:

- Застарілий інтерфейс, що не відповідає сучасним стандартам і може створювати труднощі у сприйнятті інформації.
- Обмежений набір функцій, що не дозволяє повноцінно відтворити реальний процес налаштування BIOS.
- Некоректне управління: один із симуляторів працював виключно з мишею, що суперечить реальним BIOS.

– Відсутність системи збереження змін: після оновлення сторінки всі внесені параметри скидаються, що робить симулятор малокорисним у навчальному процесі.



Рисунок 2.2 – Аналог BIOS №2

На відміну від цих рішень, розроблений симулятор BIOS Setup Utility для материнської плати ASRock 760GM-HDV є значно більш досконалим та сучасним. Його ключові переваги включають:

Реалістичний та сучасний дизайн, який відповідає актуальним версіям BIOS і не виглядає застарілим.

Повноцінне керування клавіатурою, що повністю повторює принципи взаємодії з реальними BIOS.

Розширений функціонал, який дозволяє змінювати різні параметри, зокрема системний час, частоту процесора, швидкість вентилятора, параметри пам'яті тощо.

Можливість збереження змін через LocalStorage, що дозволяє зберігати внесені налаштування та продовжувати роботу без втрати даних.

Система навчальних завдань, що мотивує користувачів виконувати дії в BIOS, отримуючи за це бали, що сприяє кращому засвоєнню матеріалу.

Завдяки цим особливостям симулятор є набагато більш ефективним інструментом для навчання, ніж знайдені аналоги. Він дозволяє користувачам отримати практичний досвід роботи з BIOS у безпечному середовищі без ризику вплинути на реальні налаштування комп'ютера.

2.4 Використання LocalStorage для збереження налаштувань

Одним із головних аспектів реалізації проєкту є ефективне керування даними, можливість їх збереження та взаємодія з інтерфейсом у реальному часі. Оскільки симулятор призначений для роботи без серверної частини, всі обчислення та збереження налаштувань здійснюються безпосередньо в браузері.

Для збереження змін у параметрах BIOS використано LocalStorage, що дозволяє зберігати інформацію без необхідності в додаткових базах даних або серверному сховищі. Це рішення є оптимальним для роботи з локальними даними, оскільки воно забезпечує миттєвий доступ до налаштувань без затримок, які можуть виникати при зверненні до сервера. Використано методи `setItem()` та `getItem()`, що дозволяє легко зчитувати та записувати дані у вигляді JSON-об'єкта.

Інтерфейс симулятора реалізований за допомогою HTML, CSS і JavaScript, що дозволяє забезпечити зручність у користуванні та відтворити автентичний вигляд BIOS. CSS використовується для стилізації елементів, зокрема для імітації синього фону та білого тексту, характерного для інтерфейсу BIOS. Окрім цього, застосовано стилі для відображення активних елементів, що допомагає користувачеві швидко орієнтуватися у віртуальному середовищі.

Обробка подій клавіатури є одним із найважливіших аспектів роботи симулятора. Використано метод `addEventListener()`, що відстежує натискання клавіш і виконує відповідні дії. Зокрема, клавіші `ArrowUp` та `ArrowDown`

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

відповідають за переміщення між пунктами меню, Enter підтверджує вибір, а Esc використовується для виходу з підменю без збереження змін.

Для створення динамічного інтерфейсу використано маніпуляції з DOM. Зміна текстових значень, відображення нових параметрів і оновлення елементів здійснюється за допомогою методів `document.getElementById()`, `querySelector()`, `textContent` та `innerHTML`. Це забезпечує швидке оновлення інтерфейсу без перезавантаження сторінки, що є критично важливим для створення реалістичної симуляції.

Для покращення взаємодії з користувачем реалізовано систему повідомлень. Використано бібліотеку `Toast.js`, що дозволяє створювати спливаючі повідомлення про дії користувача. Наприклад, при успішному збереженні налаштувань з'являється повідомлення про те, що зміни збережено, а у випадку помилки – відповідне попередження.

Реалізація системи завдань є важливим елементом освітньої частини симулятора. У межах цього механізму користувач отримує завдання, що передбачають зміну параметрів BIOS. Після виконання завдання перевіряється правильність дій, і якщо зміни внесено коректно, нараховуються бали. Це дозволяє користувачам не просто знайомитися з інтерфейсом BIOS, а й активно взаємодіяти з ним, набуваючи практичних навичок.

2.5 Система завдань у симуляторі BIOS

У симуляторі BIOS є система завдань, яка дозволяє користувачам виконувати різні операції, що імітують налаштування BIOS.

Кожне завдання має свою мету та інструкції, що дозволяє користувачам вивчати основні функції BIOS без ризику пошкодження реальних налаштувань системи. Кожне завдання приносить 10 балів, що дозволяє користувачу відслідковувати свій прогрес та успішність виконання завдань.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

Основним елементом є різноманітність завдань, що включають базові операції, такі як зміна часу і дати в BIOS, налаштування частоти процесора або швидкості вентилятора. Завдяки таким завданням користувачі мають можливість ознайомитися з основними параметрами BIOS і зрозуміти їх вплив на роботу комп'ютера. Це дає змогу практично вивчати основи налаштувань BIOS, не ризикуючи пошкодити реальні налаштування системи.

Завдання реалізовані таким чином, що користувач може керувати налаштуваннями лише за допомогою клавіатури, що є аналогом справжнього BIOS, де використовується клавіатурне управління для навігації по меню та зміни параметрів. Таким чином, користувач може виконувати завдання, як у реальному BIOS, зберігаючи інтерфейс максимально інтуїтивно зрозумілим.

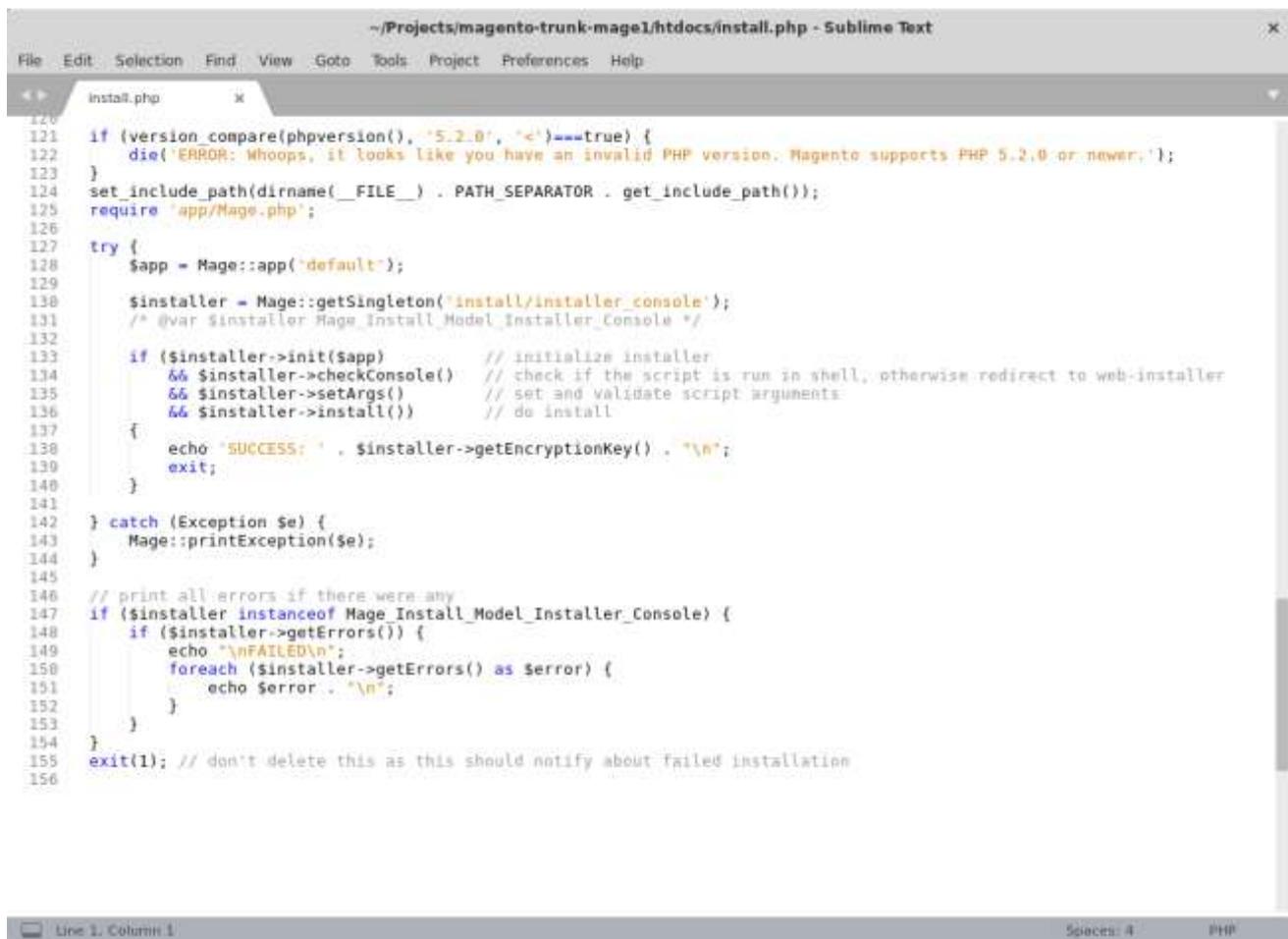
Також реалізовано функціонал збереження і завантаження користувацьких налаштувань через LocalStorage. Це дозволяє користувачу зберігати свої налаштування на випадок, якщо потрібно відновити їх після наступного входу в систему або просто для того, щоб зберегти конфігурацію для подальшого використання.

Завдання є частиною освітнього процесу, і їх виконання допомагає користувачам краще розуміти функції BIOS, покращувати свої знання та навички роботи з налаштуваннями комп'ютера. Вони дають можливість поступово опанувати всі основні функції BIOS, що в свою чергу допомагає студентам та учням глибше зрозуміти роботу комп'ютера та його налаштування.

2.6 Вибір редактора коду

Для цього проєкту було використано Visual Studio Code (рис. 2.3), оскільки він надає широкий функціонал для зручного написання коду. Однією з головних переваг цього редактора є можливість встановлення розширень, які значно полегшують процес розробки. Серед таких розширень особливо важливим став Live Server (рис. 2.7), який дозволяє запускати проєкт на локальному сервері. Це

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27



```
~/Projects/magento-trunk-mage1/htdocs/install.php - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
install.php x
120
121 if (version_compare(PHP_VERSION(), '5.2.0', '<')==true) {
122     die('ERROR: Whoops, it looks like you have an invalid PHP version. Magento supports PHP 5.2.0 or newer.');
```

Рисунок 2.4 - Sublime Text

Notepad++ – це ще один популярний редактор коду, який має невеликий розмір і швидку роботу (рис. 2.5). Він підтримує багато мов програмування, має мінімалістичний інтерфейс і невисокі системні вимоги. Проте його функціонал значно обмежений у порівнянні з Visual Studio Code, оскільки в ньому відсутня глибока інтеграція з розширеннями та інструментами розробки, такими як Git або вбудований термінал.

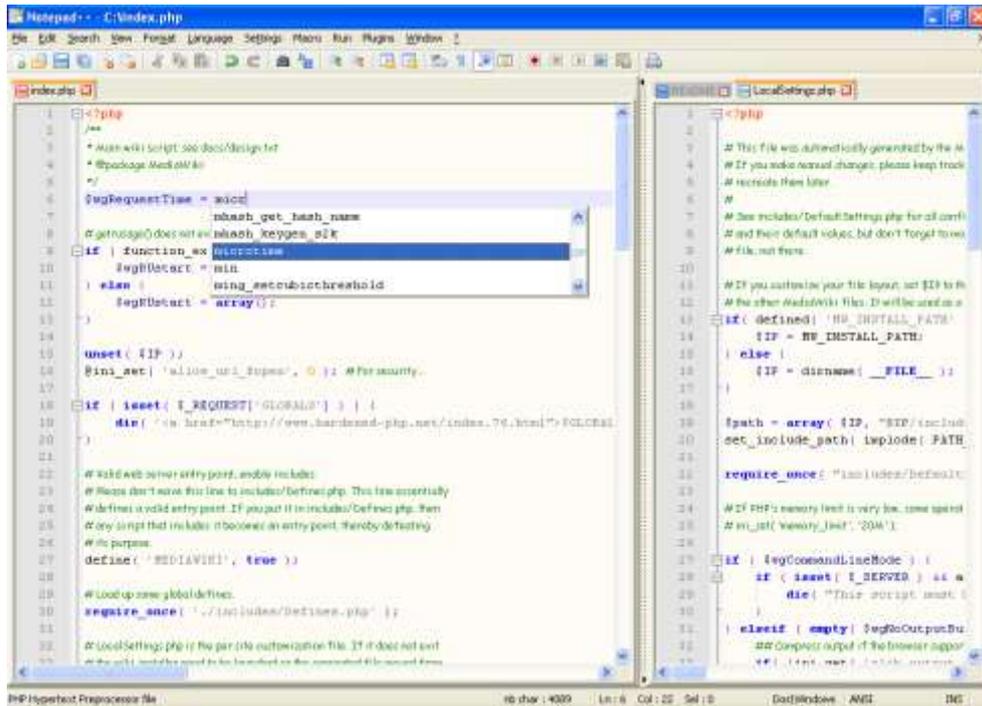


Рисунок 2.5 - Notepad++

Atom – це редактор, створений компанією GitHub, який має схожу концепцію з Visual Studio Code. Він підтримує велику кількість розширень, налаштовуваний інтерфейс та інтеграцію з Git. Проте він поступається у швидкодії, споживає більше системних ресурсів і працює повільніше на слабких комп'ютерах.

atom/atom-dark-ui

The default dark ui theme for Atom



23
Contributors

0
Issues

103
Stars

410
Forks



Рисунок 2.6 – Atom

									Арк.
									30
Змн.	Арк.	№ докум.	Підпис	Дата					

ДП.123.211.014.00 ПЗ

Основною причиною вибору Visual Studio Code стало поєднання швидкодії, розширеного функціоналу, інтеграції з Git, можливості роботи з Live Server (рис. 2.7), а також підтримка великої кількості корисних розширень.

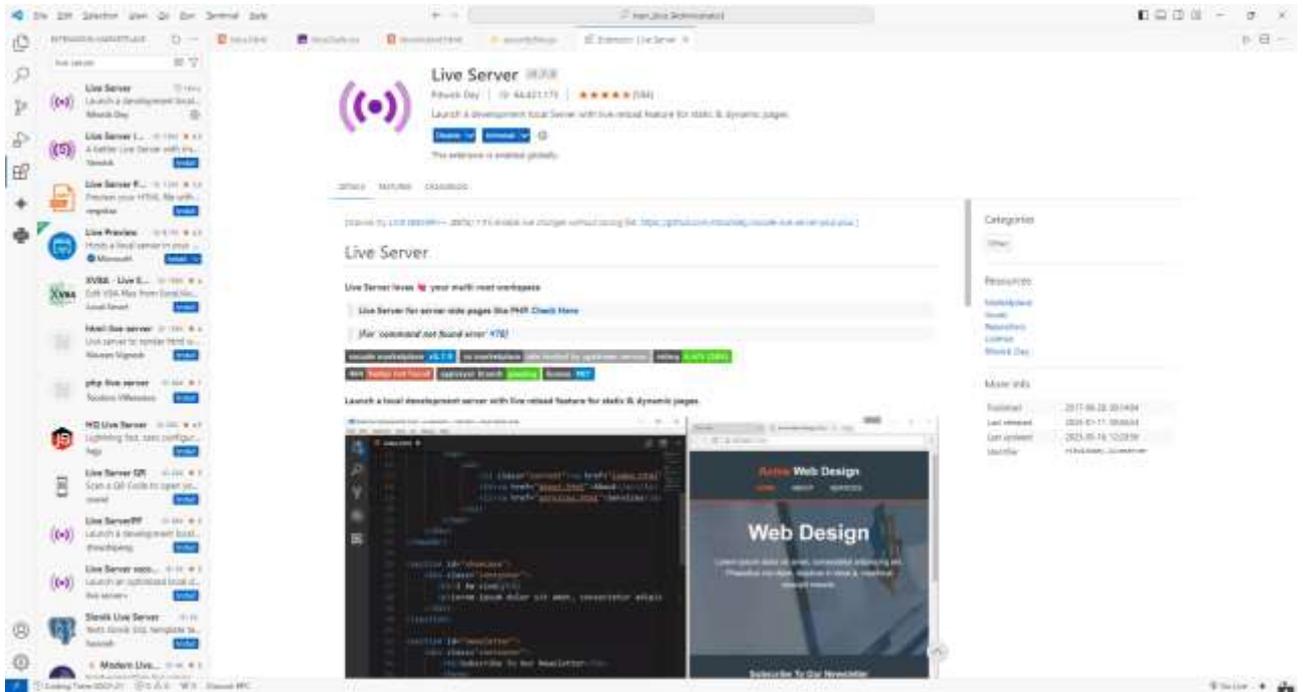


Рисунок 2.7 – Розширення Live Server

Його перевагою є кросплатформеність, завдяки чому редактор можна використовувати як на Windows, так і на Linux або macOS. Додатково, вбудована функція IntelliSense допомагає під час написання коду, пропонуючи автозаповнення та підказки, що значно полегшує процес розробки.

Таким чином, Visual Studio Code став найкращим вибором для цього проекту, оскільки забезпечив зручний та ефективний процес роботи з кодом, спрощене тестування та швидке внесення змін.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31

3 СПЕЦІАЛЬНА ЧАСТИНА

3.1 Загальний вид програми

Головне меню програми складається з декількох основних розділів, кожен з яких виконує певні функції та містить налаштування, необхідні для конфігурації системи. Вікно програми має такий інтерфейс: Main, OC Tweaker, Advanced, H/W Monitor, Boot, Security, Exit і Завдання (рис. 3.1).



Рисунок 3.1 – Загальний вид

На рисунку 3.2 представлена діаграма компонентів відображає структурну архітектуру програмного забезпечення симулятора BIOS. Вона ілюструє основні логічні модулі системи, їх взаємозв'язки та залежності. Така діаграма дозволяє краще зрозуміти, як побудований вебсайт, які компоненти відповідають за конкретний функціонал, та як між собою взаємодіють окремі частини проекту.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

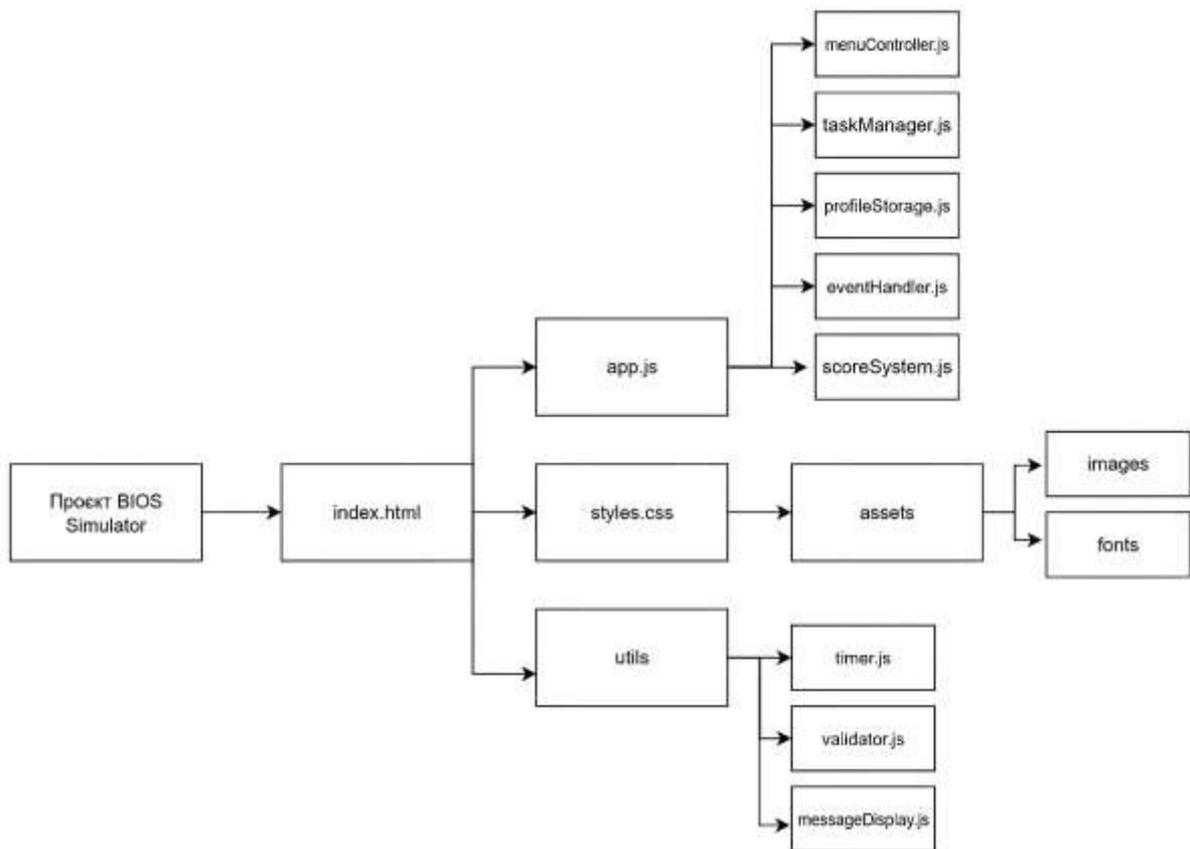


Рисунок 3.2 – Діаграма компонентів

У розділі Main (рис. 3.1) знаходиться базова інформація про систему, включаючи поточну дату та час, модель процесора, об'єм оперативної пам'яті та інші загальні характеристики обладнання.

ОС Tweaker (рис. 3.3) містить параметри розгону процесора та пам'яті. У цьому розділі можна змінювати множник процесора, напругу живлення, таймінги пам'яті та інші параметри, що впливають на продуктивність системи.



Рисунок 3.3 – OC Tweaker

Advanced (рис. 3.4) надає доступ до більш глибоких налаштувань обладнання.



Рисунок 3.4 – Advanced

Він містить декілька підпунктів:

1) CPU Configuration – дозволяє змінювати налаштування процесора, такі як Cool' n' Quiet, Secure Virtual Machine, Enhanced Halt State (C1E) та CPU Thermal Throttle. Ці параметри відповідають за енергозбереження, віртуалізацію та тепловий контроль процесора.

2) Chipset Configuration – містить налаштування чіпсета, які впливають на роботу внутрішніх компонентів материнської плати.

3) ACPI Configuration – дозволяє змінювати параметри керування живленням і режимами сну.

4) Storage Configuration – містить налаштування жорстких дисків, режимів SATA (AHCI, IDE, RAID) та інших параметрів збереження даних (рис. 3.5).



Рисунок 3.5 - Storage Configuration

5) PCIPnP Configuration – відповідає за налаштування підключених PCI-пристроїв.

6) USB Configuration – містить налаштування USB-контролера та режимів роботи USB-портів.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

7) Instant Flash – утиліта для оновлення прошивки BIOS з USB-накопичувача.

8) H/W Monitor (рис. 3.6) відповідає за моніторинг стану обладнання. У цьому розділі можна переглядати температуру процесора, швидкість обертання вентиляторів та інші параметри.



Рисунок 3.6 - H/W Monitor

Він містить підпункти:

9) CPU Fan Setting – дозволяє змінювати швидкість обертання кулера процесора залежно від температури (рис. 3.7).



Рисунок 3.7 - CPU Fan Setting

10) Chassis Fan 1 Setting та Chassis Fan 2 Setting – подібні параметри для корпусних вентиляторів.

11) OTP (Over Temperature Protection) – функція захисту від перегріву, яка автоматично вимикає систему при критичному підвищенні температури.

12) Case Open Feature – параметр, який відстежує, чи відкривали корпус комп'ютера.

Boot містить налаштування завантаження системи. У цьому розділі можна вибрати пріоритет завантажувальних пристроїв, вмикати або вимикати швидке завантаження та змінювати інші параметри, що впливають на процес старту операційної системи.

Security відповідає за налаштування безпеки. У цьому розділі можна встановлювати або змінювати паролі BIOS, включаючи Change Supervisor Password (пароль адміністратора) та пароль користувача (рис. 3.11). Це дозволяє захистити доступ до налаштувань BIOS від несанкціонованих змін.

Exit містить опції виходу з BIOS, збереження або скасування внесених змін, а також можливість завантажити стандартні налаштування за замовчуванням.

					ДП.123.211.014.00 ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

Окремий розділ Завдання містить систему симуляції завдань (рис. 3.15), які користувач може виконувати для вивчення BIOS. Завдання пов'язані зі зміною певних параметрів, таких як встановлення часу, зміна частоти процесора або встановлення пароля адміністратора. Виконання цих завдань дозволяє на практиці освоїти основні функції BIOS у безпечному середовищі без ризику змін у реальній системі.

3.2 Можливість збереження та завантаження профілів користувачів

Збереження та завантаження профілів користувачів є однією з ключових функцій онлайн-симулятора BIOS, що забезпечує зручність використання та дозволяє користувачам зберігати свої налаштування між сесіями. Ця функція дозволяє користувачам не тільки зберігати всі зміни, які вони вносять до BIOS, але й повернутися до них пізніше, не ризикуючи втратити важливі конфігурації. Збереження профілів також сприяє більш ефективному освітньому процесу, оскільки дозволяє зберігати прогрес і легко повертатися до вже виконаних завдань та налаштувань.

Однією з основних переваг цієї функції є можливість збереження профілю на локальному комп'ютері через механізм LocalStorage. Використовуючи LocalStorage, всі зміни, зроблені користувачем у симуляторі BIOS, зберігаються без необхідності наявності додаткових серверів чи баз даних. Це забезпечує простоту та швидкість збереження даних, а також робить симулятор доступним без залежності від наявності підключення до Інтернету. Користувач може зберегти профіль з усіма налаштуваннями в одне натискання і йому покажеться повідомлення про те, що він успішно зберіг профіль, та через деяку годину вікно саме закриється, знизу під текстом є смужка, по ній можна зрозуміти через скільки закриється повідомлення (рис. 3.10).

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		38

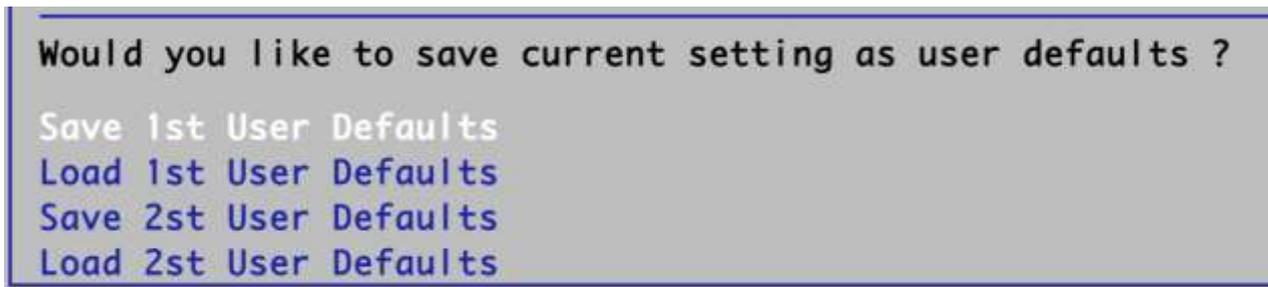


Рисунок 3.8 – Параметри збереження та завантаження профілю

Функція збереження профілю забезпечує високий рівень персоналізації. Користувач може налаштувати BIOS відповідно до своїх уподобань і зберегти ці налаштування для майбутніх сесій. Це особливо корисно, якщо користувач працює з кількома різними конфігураціями або проектами. Наприклад, під час навчання в симуляторі можна створити профіль для кожної з типових задач, таких як налаштування часу, зміна швидкості процесора чи вентиляторів. Завантаживши профіль, користувач може швидко повернутися до потрібної конфігурації, не витрачаючи час на повторне налаштування параметрів (рис. 3.8).

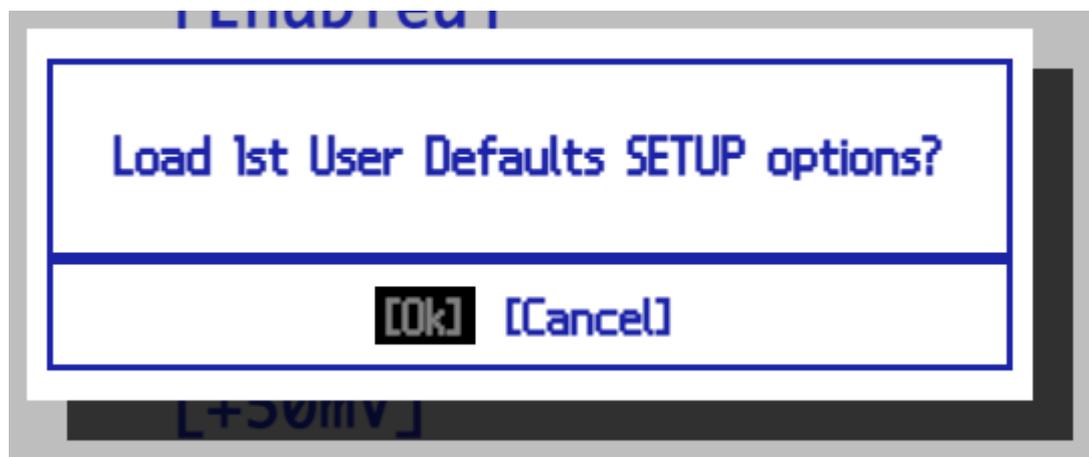


Рисунок 3.9 – Повідомлення про завантаження профілю

Також важливим аспектом є можливість одночасного збереження кількох профілів. Користувач може створювати кілька профілів з різними налаштуваннями та зберігати їх для подальшого використання. Це дозволяє,

наприклад, мати один профіль для стандартних налаштувань BIOS, інший — для більш агресивних налаштувань, таких як розгін процесора. Користувач може безпечно змінювати ці налаштування, знаючи, що в будь-який час можна повернутися до попереднього профілю.

Процес завантаження профілю працює так само просто, як і збереження. Користувач може вибрати збережений профіль і завантажити його в симулятор, після чого всі налаштування BIOS будуть автоматично відновлені. Це дозволяє значно заощадити час, оскільки не потрібно вручну налаштовувати

параметри кожного разу при запуску симулятора. Крім того, завантаження профілю гарантує, що користувач завжди працюватиме з тією самою конфігурацією, що і при попередньому використанні.

Важливою частиною цієї функції є також механізм безпеки. Завантаження профілю здійснюється тільки після підтвердження користувача, що запобігає випадковому заміненню поточних налаштувань (рис. 3.9). Якщо користувач не бажає замінювати поточний профіль, йому надається можливість скасувати завантаження і продовжити роботу з поточними налаштуваннями. Цей підхід забезпечує додаткову гнучкість і контроль над процесом збереження та завантаження профілів.

Інтерфейс для роботи з профілями простий і зрозумілий. Користувач може легко створити новий профіль, зберегти його або завантажити вже існуючий. Після того як профіль збережений, всі налаштування і параметри, що були змінені, автоматично зберігаються в LocalStorage, і їх можна відновити навіть після закриття браузера. Всі ці функції створюють надійний механізм збереження прогресу, що дозволяє користувачам продовжувати свою роботу з того місця, де вони зупинилися.

					ДП.123.211.014.00 ПЗ	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		40

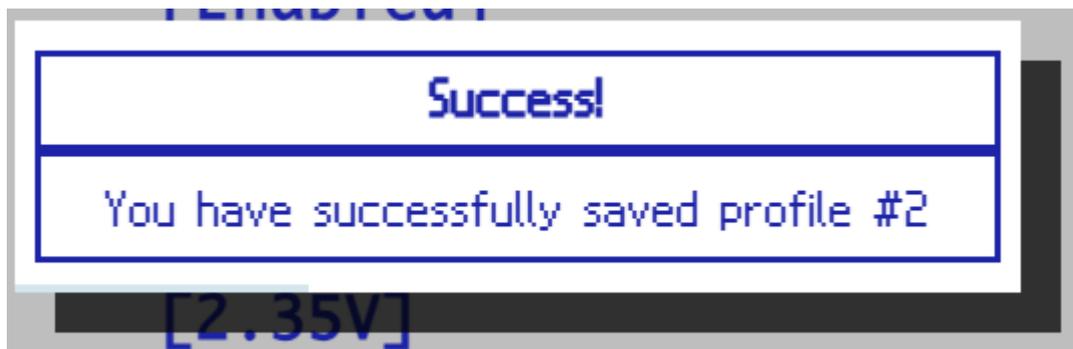


Рисунок 3.10 – Повідомлення про збереження профілю

Функціональність збереження та завантаження профілів користувачів також має практичне значення для освітніх установ (рис. 3.9). Студенти можуть використовувати цей симулятор для вивчення різних конфігурацій BIOS без необхідності постійно змінювати налаштування на реальному комп'ютері.

Завдяки збереженню профілів можна створювати спеціальні навчальні програми та курси, які включають різні налаштування та завдання, що дозволяє ефективно навчати учнів без ризику пошкодження реальних систем.

Завдяки такій функціональності симулятор BIOS не тільки стає зручним інструментом для навчання, але й надає можливість створення індивідуальних налаштувань, які можуть бути адаптовані під різні потреби користувачів.

3.3 Система паролів для доступу до налаштувань

Система паролів для доступу до налаштувань у симуляторі BIOS забезпечує базовий рівень безпеки для користувачів, дозволяючи встановлювати обмежений доступ до конфігурації. У симуляторі реалізовано два типи паролів: User Password і Supervisor Password. Основна мета цієї системи – продемонструвати принцип роботи паролів у BIOS, а також надати можливість користувачам навчитися їх налаштовувати та змінювати (рис. 3.12).

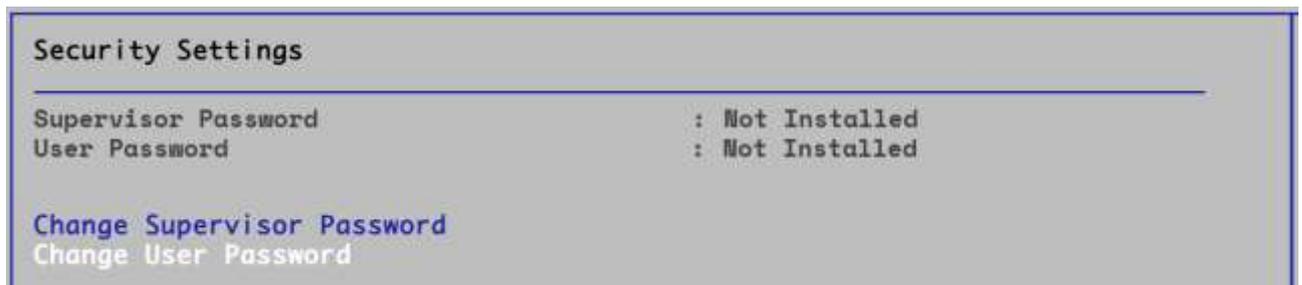


Рисунок 3.11 – Параметри паролю

Коли користувач обирає Change User Password, відкривається вікно введення пароля, в якому потрібно ввести комбінацію символів. Якщо пароль містить менше ніж три символи, зміни не застосовуються, і система не реагує на введене значення (рис. 3.13).

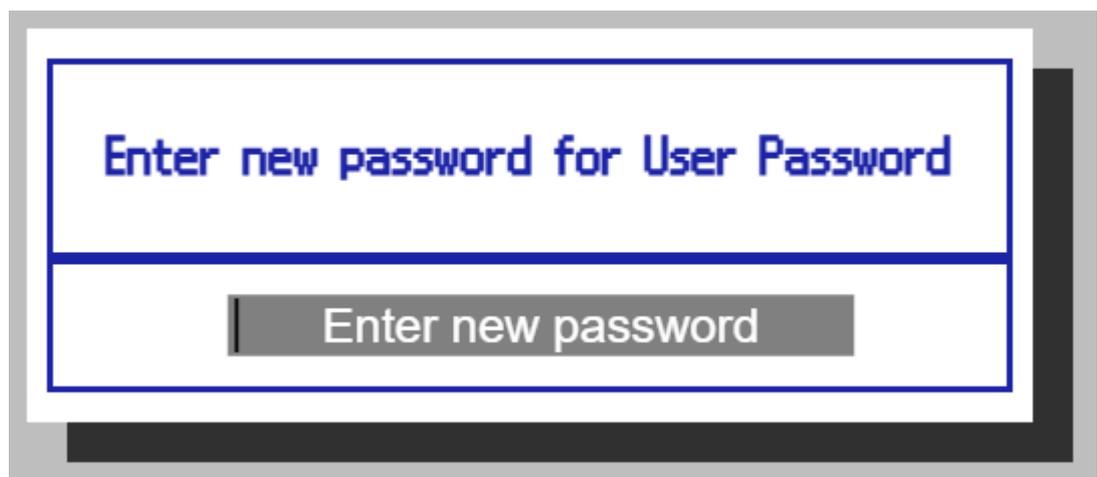


Рисунок 3.12 – Введення паролю User Password

У разі введення чотирьох або більше символів пароль успішно встановлюється, і статус User Password : Not Installed змінюється на Installed. Це означає, що тепер при вході в симулятор потрібно буде ввести збережений пароль, щоб отримати доступ до налаштувань (рис. 3.13).



Рисунок 3.13 – Успішне встановлення паролю

Функція зміни пароля реалізована таким чином, що дозволяє зберігати введений користувачем пароль у локальному сховищі браузера. Це забезпечує збереження налаштувань навіть після закриття сторінки або перезапуску браузера. Однак у разі видалення кешу або даних сайту всі збережені паролі будуть скинуті.

Крім можливості встановлення пароля, користувач може його скинути. Для цього необхідно обрати Change User Password і натиснути клавішу F9.

На екрані з'явиться діалогове вікно з варіантами OK та Cancel. Якщо користувач обирає OK, пароль видаляється, і статус змінюється назад на Not Installed. Якщо натиснуто Cancel, жодних змін не відбувається, і поточний пароль залишається активним (рис. 3.14).

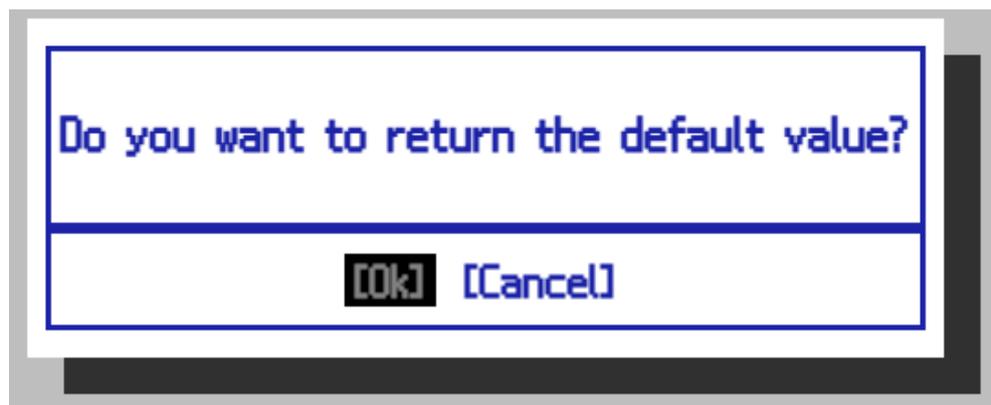


Рисунок 3.14 – Скидання паролю

Supervisor Password має аналогічний функціонал, дозволяючи встановлювати додатковий пароль. Однак цей пароль не має реального впливу на

систему, його можна лише встановити або видалити. Його наявність у симуляторі демонструє на практиці, як у реальному BIOS можна встановлювати два рівні паролів: один для звичайних користувачів і один для адміністратора.

Механізм роботи паролів у симуляторі спрощений і адаптований для навчальних цілей. Реалізація базується на використанні JavaScript та маніпуляції DOM-елементами для відображення змін у статусі пароля. Використання локального сховища дозволяє зберігати встановлені паролі між сеансами без необхідності підключення до бази даних або сервера.

Система паролів у симуляторі сприяє кращому розумінню того, як працює захист BIOS у реальних комп'ютерах. У реальному BIOS паролі можуть обмежувати доступ до критичних налаштувань або навіть блокувати завантаження системи без правильного введення коду. У симуляторі ці обмеження не реалізовані, але сама можливість встановлення і скидання паролів дає користувачам змогу навчитися працювати з цією функцією без ризику втрати доступу до реального комп'ютера.

Крім основних функцій встановлення та видалення пароля, симулятор дозволяє користувачам експериментувати з параметрами без ризику блокування доступу. У реальному середовищі встановлення пароля може призвести до необхідності скидання налаштувань материнської плати, якщо користувач забуде встановлений пароль. У симуляторі ж користувач може в будь-який момент видалити пароль через меню без додаткових дій.

3.4 Система завдань для покращення навичок роботи з BIOS

Система завдань у симуляторі BIOS спрямована на покращення навичок користувачів у роботі з базовими налаштуваннями системи. Вона дозволяє користувачам виконувати реальні сценарії використання BIOS, такі як зміна параметрів завантаження, налаштування швидкості обертання кулерів, розгін процесора, встановлення часу та багато інших. Кожне завдання розраховане на

					ДП.123.211.014.00 ПЗ	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		44

конкретну дію, яку користувач має виконати, що допомагає краще засвоїти функціонал BIOS (рис. 3.15).

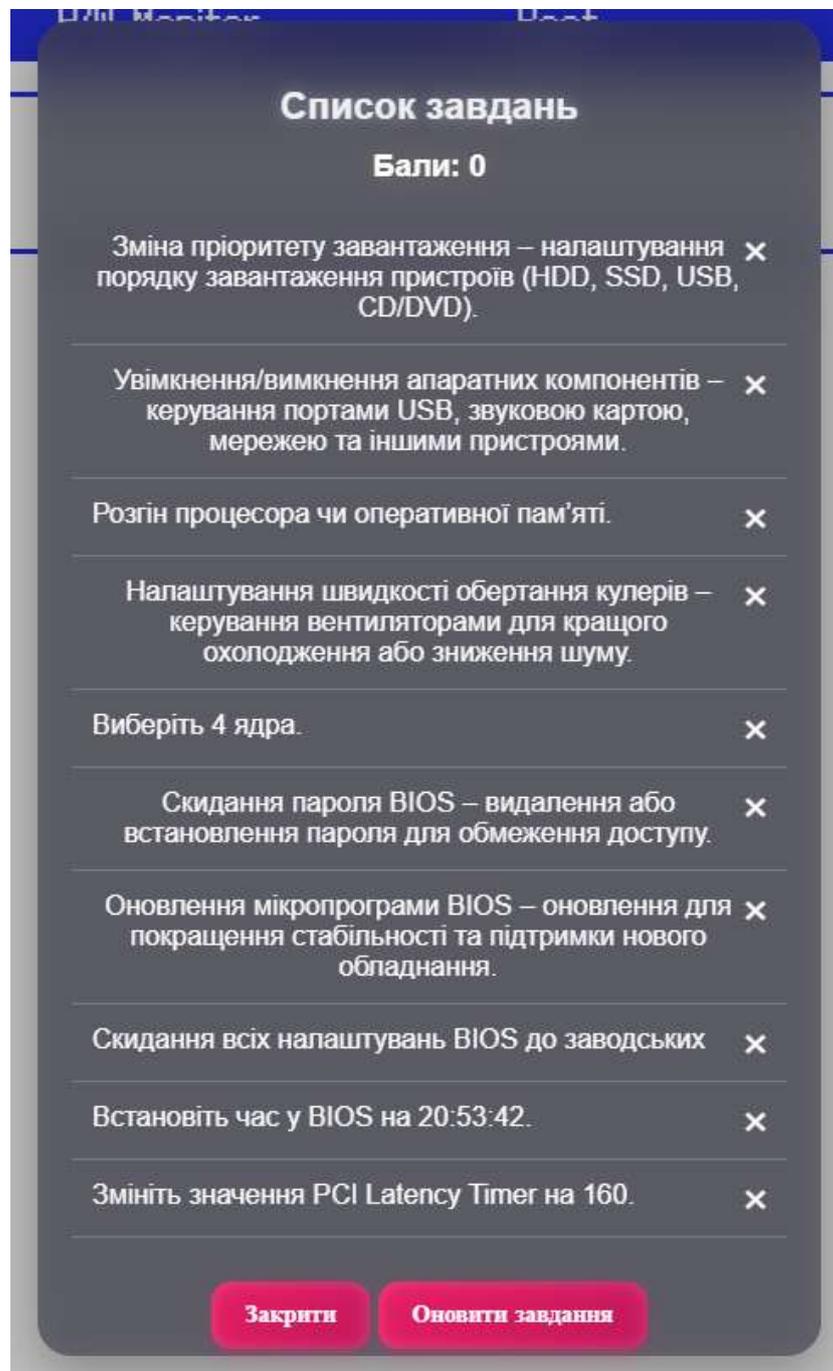


Рисунок 3.15 – Завдання

Завдання доступні через окремий інтерфейс, який знаходиться на панелі меню, на нього потрібно натиснути. Після відкриття списку користувач може

					ДП.123.211.014.00 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

обрати будь-яке завдання та спробувати його виконати. За успішне виконання кожного завдання нараховується 10 балів. Праворуч від назви завдання є статус виконаного завдання, хрестик означає, що його не виконано, а галочка виконано, значить, що завдання зроблено. Це дозволяє користувачам не лише навчитися працювати з BIOS, а й отримати певну мотивацію до проходження всіх можливих сценаріїв.

Система завдань включає різноманітні сценарії. Наведемо деякі з них. Зміна пріоритету завантаження – користувач повинен налаштувати порядок завантаження пристроїв, вибираючи між HDD, SSD, USB або CD/DVD. Це корисна навичка, яка допомагає у випадках, коли потрібно встановити операційну систему або запустити комп'ютер з альтернативного джерела. Вимкнення або увімкнення апаратних компонентів – користувач може спробувати змінити стан USB-портів, звукової карти або мережевого адаптера, що дозволяє зрозуміти принципи керування периферійними пристроями. Розгін процесора або оперативної пам'яті – завдання включає зміну параметрів частоти роботи процесора або оперативної пам'яті, що дозволяє зрозуміти основи оверклокінгу. Налаштування швидкості обертання кулерів – користувач може регулювати швидкість вентиляторів для оптимального охолодження або зменшення рівня шуму.

Усі завдання базуються на реальних функціях BIOS, що дозволяє користувачам отримати практичний досвід. Вони виконуються в інтерактивному режимі, а перевірка правильності здійснюється через систему логіки, яка аналізує зміни в параметрах BIOS. Якщо користувач вірно змінив налаштування, система автоматично зараховує виконане завдання і нараховує 10 балів.

Щоб забезпечити зручність користувачів, система завдань реалізована таким чином, що вона не впливає на сам функціонал BIOS, а є окремим інтерактивним елементом. Завдяки цьому користувач може у будь-який момент переглянути список доступних завдань і спробувати виконати будь-яке з них без ризику втратити важливі зміни.

					ДП.123.211.014.00 ПЗ	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		46

Система завдань створена таким чином, щоб навіть початківці могли швидко зрозуміти принципи роботи BIOS. Вона дозволяє експериментувати з налаштуваннями, спостерігати за їхніми змінами та вчитися приймати правильні рішення в умовах, наближених до реальних. Оскільки завдання не потребують введення спеціальних команд або складних операцій, вони доступні для широкої аудиторії, включаючи школярів і студентів, які тільки починають вивчати принципи роботи комп'ютерів.

Завдяки інтерактивному підходу та системі оцінювання користувачі можуть відслідковувати свій прогрес. Це створює ефект гри, що додатково мотивує проходити більше завдань і покращувати свої навички. У результаті система завдань у симуляторі BIOS є ефективним засобом навчання, який допомагає зрозуміти основи налаштування комп'ютера без ризику змінити важливі параметри в реальному середовищі.

3.5 Навігація та управління

Навігація та управління через клавіатуру у симуляторі BIOS відіграють ключову роль, оскільки саме такий спосіб взаємодії є стандартним для реального BIOS. Для цього проєкта було реалізовано систему перемикання між елементами меню, вибору опцій та введення значень через клавіатуру, що дозволяє користувачеві повністю зануритися в процес роботи з BIOS.

Основне управління здійснюється за допомогою клавіш стрілок (←, →, ↑, ↓), які відповідають за переміщення між розділами та елементами налаштувань. Перемикання між основними вкладками BIOS (наприклад, Main, Advanced, Boot, Exit) здійснюється клавішами ← та →, тоді як переміщення між параметрами у вибраному розділі виконується за допомогою клавіш ↑ та ↓ (рис. 3.16). Це дозволяє користувачеві швидко орієнтуватися в меню та змінювати потрібні параметри.

Для підтвердження вибору або переходу до редагування певного параметра використовується клавіша Enter.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

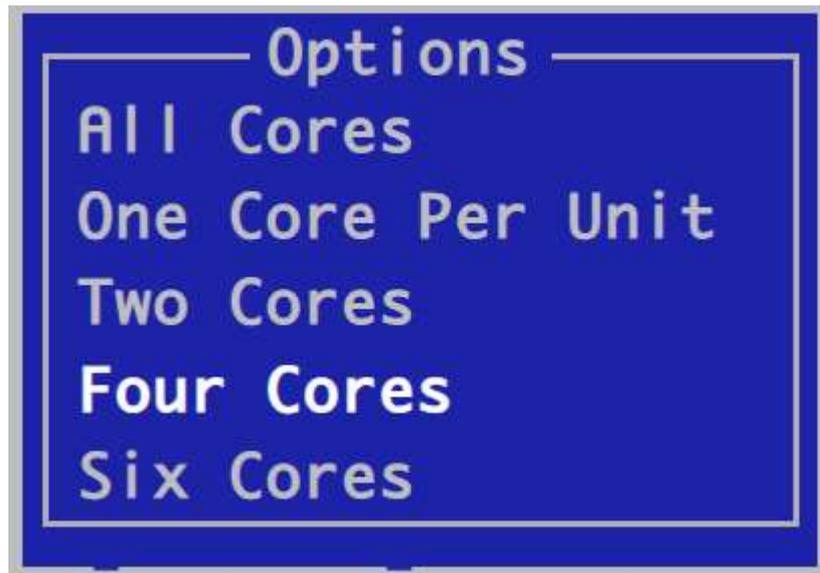


Рисунок 3.16 – Вибір параметрів

При натисканні цієї клавіші відкривається випадаючий список (якщо доступний вибір з декількох варіантів) або з'являється поле введення, де користувач може змінити значення вручну. У деяких випадках (наприклад, при зміні часу або введенні числових значень) активується можливість редагування параметра безпосередньо, після чого зміни підтверджуються натисканням Enter.

Для скидання значень до заводських налаштувань у симуляторі передбачена клавіша F9. При її натисканні з'являється вікно підтвердження з можливістю вибору OK або Cancel (рис. 3.17), що дозволяє уникнути випадкових змін. Це відповідає логіці роботи реального BIOS, де також передбачені механізми відновлення стандартних параметрів.

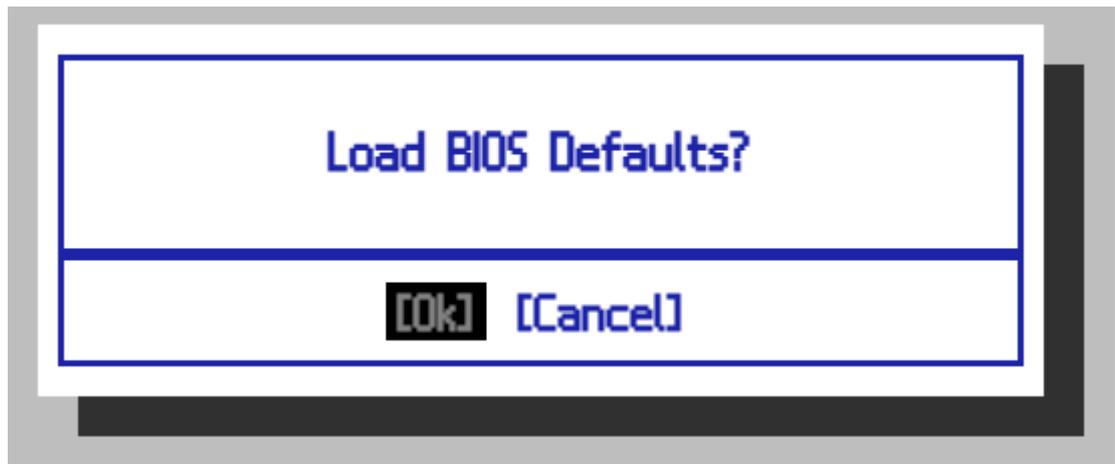


Рисунок 3.17 – Заводські налаштування

Клавіша Esc використовується для виходу з поточного розділу або повернення на попередній рівень меню. Якщо користувач знаходиться в головному меню, натискання Esc може викликати запит на вихід з BIOS, що додає реалістичності симулятору. Завдяки цьому користувач отримує досвід роботи з BIOS, максимально наближений до справжнього середовища.

Особлива увага приділяється логіці перемикання між елементами меню. Для цього проекту було реалізовано функції switchMenuItem та switchLine, які дозволяють переміщуватися між вкладками та параметрами без помилок. Активний елемент меню зберігається у змінній activeMenuItemIndex, яка оновлюється залежно від натискання клавіш навігації. Завдяки цьому забезпечується коректне переміщення між розділами BIOS без виходу за межі доступних опцій.

Переключення між рядками всередині активного розділу реалізується через функцію switchLine, яка враховує лише видимі елементи. Це дозволяє уникнути ситуацій, коли користувач випадково натискає клавішу вниз або вгору, а курсор переходить на порожнє місце. Усі зміни відображаються в реальному часі, що забезпечує плавну навігацію без затримок.

Крім стандартного управління, у симуляторі передбачена можливість зміни деяких параметрів за допомогою клавіш + та -. Це особливо корисно для налаштувань, які підтримують числові значення або мають фіксований набір

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

параметрів (наприклад, вибір значення PCI Latency Timer). Оскільки для реального BIOS характерний мінімалістичний інтерфейс без графічних елементів, система управління через клавіатуру створює відчуття роботи з автентичним середовищем. Усі дії користувача обробляються через події клавіатури, що дозволяє миттєво реагувати на введення команд. Завдяки цьому симулятор надає максимально реалістичний досвід взаємодії з BIOS, допомагаючи користувачам навчитися працювати з системними налаштуваннями.

3.6 Дрібні функціональні можливості

У рамках проєкту реалізовано низку дрібних функціональних можливостей, які сприяють більшій інтерактивності симулятора BIOS та створюють ефект справжнього середовища BIOS. Однією з таких можливостей є візуальне підсвічування активного елемента меню. Коли користувач переміщується між пунктами за допомогою клавіш зі стрілками вгору або вниз, до активного елемента застосовується клас active, що змінює його колір на білий. Це дозволяє легко орієнтуватися в інтерфейсі, що імітує поведінку реального BIOS (рис. 3.18).

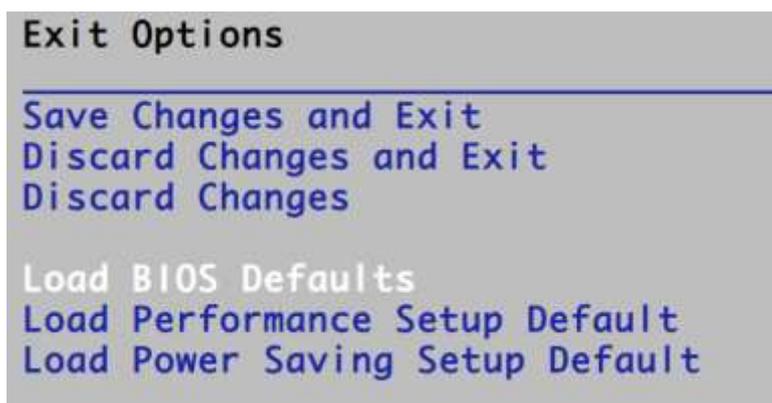


Рисунок 3.18 – Підсвічування активного елемента

Ще одним важливим аспектом є можливість виходу з налаштувань без збереження змін. Якщо користувач відкрив певний параметр і передумав його змінювати, він може просто натиснути клавішу ESC. Це закриє вікно

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

налаштувань, залишивши всі параметри без змін, що також відповідає поведінці реального BIOS.

Анімацій у базовому інтерфейсі BIOS немає, оскільки оригінальне середовище також не містить візуальних ефектів. Однак у вкладці "Завдання" реалізовані анімації, що додають динамічності роботі з цією частиною симулятора.

Змінювати мову інтерфейсу неможливо, що є ще одним елементом, який повторює справжню поведінку BIOS. У більшості реальних BIOS користувач не має можливості змінювати мову меню, тому цей функціонал також відсутній у симуляторі.

Система збереження налаштувань працює через клавішу F10. Після її натискання з'являється вікно підтвердження з варіантами OK або Cancel. Якщо користувач обирає OK, відбувається імітація виходу з BIOS: екран перемикається на сторінку завантаження, де відображається стандартний текст Press F2 or DEL to Run Setup, Press F6 for Instant Flash тощо. Після п'яти секунд очікування відбувається анімація завантаження системи, і напис змінюється на Entering SETUP..., що створює враження, ніби користувач натиснув F2 для повторного входу в BIOS. Такий підхід реалізований через те, що в рамках симулятора неможливо запустити реальну операційну систему після виходу з BIOS (рис. 3.19).

					ДП.123.211.014.00 ПЗ	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		51

```
AMIBIOS(C) 2006 American Megatrends, Inc.  
7606M-HDV BIOS P1.20  
CPU : AMD FX(tm)-8320 Eight-Core Processor  
Speed : 4.00 GHz Count : 8
```

```
Press F2 or DEL to Run Setup  
Press F6 for Instant Flash  
Press F11 for Boot Menu
```

```
Dual-Channel Memory Mode
```

```
(C) American Megatrends, Inc.  
64-111-0000010-00101111-051118-AMD-A2014111-Y2K5
```

6838

Рисунок 3.19 – Вхід в BIOS

Якщо при збереженні були вибрані неправильні параметри, через п'ять секунд очікування текст змінюється на Wrong settings..., що повідомляє користувачу про помилку. Після цього знову надається можливість увійти до BIOS для виправлення параметрів. Це є додатковою функцією, яка дозволяє відтворити логіку реальних BIOS, де невірні налаштування можуть призвести до проблем із завантаженням системи (рис. 3.20).

					ДП.123.211.014.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

```

AMIBIOS(C) 2006 American Megatrends, Inc.
7606M-HDV BIOS P1.20
CPU : AMD FX(tm)-8320 Eight-Core Processor
Speed : 4.00 GHz Count : 8

Wrong settings...
Correct the settings so that the system will start.
Press F2 to correct the settings
Press F6 for Instant Flash
Press F11 for Boot Menu

Dual-Channel Memory Mode
16368MB OK
Auto-Detecting SATAII_4.IDE Hard Disk
Auto-Detecting SATAII_1.AHCI Hard Disk
SATAII_4: TOSHIBA DT01ACA100 M520A0A0
Ultra DMA Mode-6, S.M.A.R.T. Capable but Disabled
SATAII_1: Samsung SSD 860 EVO 250GB RVT04B6Q
S.M.A.R.T. Capable and Status OK
00B1

```

Рисунок 3.20 – Неправильні параметри BIOS

Усі ці дрібні функції сприяють більш реалістичній взаємодії з симулятором та допомагають користувачу краще розуміти логіку роботи BIOS.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Витрати на оплату праці

Розробка онлайн-симулятора BIOS є складним процесом, що включає різні етапи роботи: написання коду, тестування, створення інтерфейсу користувача, налагодження функціоналу та підтримку проекту. У разі комерційного підходу до розробки важливо оцінити витрати на оплату праці спеціалістів, які були б залучені до проекту.

Основні витрати на оплату праці формуються з урахуванням таких факторів: кількість годин, витрачених на розробку, середня ринкова ставка розробників відповідного рівня, додаткові витрати на тестування, підтримку та покращення проекту. Для розрахунку можна взяти середню погодинну ставку веброзробника в Україні, яка становить 10–15\$ за годину. Припустимо, що на створення симулятора витрачено 200 годин роботи. У такому разі вартість роботи розробника складе:

- вартість=200×10=2000 (мінімальна оцінка);
- вартість=200×15=3000 (максимальна оцінка).

Таким чином, тільки на оплату праці веброзробника потрібно було б 2000–3000\$.

Якщо додати тестувальника, який перевіряє коректність роботи симулятора та усуває баги, та врахувати витрати на дизайнера, який покращує UI/UX, загальна сума може зрости ще на 500–1000\$.

Розрахунок вартості роботи та інших витрат наведено в таблиці 4.1.

					ДП.123.211.014.00 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 4.1 – Загальна вартість розробки

Категорія витрат	Години роботи	Погодинна ставка (\$)	Загальна сума (\$)
Веброзробка (JS, HTML, CSS)	200	10-15	2000-3000
Тестування	30	8-12	240-360
UI/UX-дизайн	40	12-20	480-800
Загальна вартість	-	-	2720–4160

У випадку, якщо розробка проєкту передбачала довготривалу підтримку, необхідно було б також закласти додаткові витрати на оновлення функціоналу, оптимізацію коду та виправлення можливих помилок.

Залежно від складності й вимог до проєкту, фінальна сума могла б змінюватися, але в середньому витрати на оплату праці для реалізації подібного симулятора становили б близько 3000–4000\$.

4.2 Потенційні джерела доходу та монетизація проєкту

Розробка Онлайн-симулятор BIOS може бути монетизований різними способами, враховуючи його навчальну спрямованість та потенційний попит серед користувачів. Основні джерела доходу можуть включати продаж ліцензій для навчальних закладів, впровадження преміальних функцій, рекламу та спонсорські партнерства.

Один з варіантів – продаж доступу до розширених можливостей. Наприклад, базова версія симулятора може бути безкоштовною, а розширена версія з додатковими функціями пропонуватися за підпискою. Потенційні преміум-функції: Доступ до складніших завдань та тестів з BIOS, симуляція різних моделей BIOS із розширеними параметрами, докладні аналітичні звіти про виконані завдання та помилки, індивідуальні навчальні програми.

Якщо встановити підписку на рівні 5\$ на місяць для окремих користувачів і орієнтуватися на 1000 активних користувачів, місячний дохід складе:

					ДП.123.211.014.00 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

- $5 \times 1000 = 5000$ (місячний дохід);
- $5000 \times 12 = 60000$ (річний дохід).

Навчальні заклади можуть бути зацікавлені у придбанні корпоративної ліцензії, яка надасть доступ до симулятора всім студентам. Якщо припустити, що така ліцензія коштуватиме 500\$ на рік та її придбають 20 університетів, отримаємо:

- $500 \times 20 = 10000$ (річний дохід від освітніх установ).

Ще один варіант – розміщення реклами. Якщо домовитися про партнерство з компаніями, що займаються продажем комп'ютерного обладнання або програмного забезпечення, вони можуть розміщувати рекламу на сторінках симулятора. Припустимо, що середній рекламний дохід становить 300\$ на місяць:

- $300 \times 12 = 3600$ (річний дохід від реклами).

Орієнтовний розрахунок потенційного доходу наведено у таблиці 4.2:

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

Таблиця 4.2 – Орієнтовний розрахунок потенційного доходу

Джерело доходу	Разова/Періодична оплата	Очікувана кількість покупців	Місячний дохід (\$)	Річний дохід (\$)
Підписка користувачів	5\$/місяць	1000	5000	60000
Ліцензії для університетів	500\$/рік	20	-	10000
Реклама	300\$/місяць	-	300	3600
Загальний дохід	-	-	5300	73600

Таким чином, при правильній монетизації проєкт може приносити дохід у діапазоні 70 000 – 75 000\$ на рік, що робить його потенційно вигідним з комерційної точки зору.

5 ОХОРОНА ПРАЦІ

5.1 Загальні положення

До виконання робіт з монтажу силових та освітлювальних кабельних мереж допускаються робітники, які досягли 18 років та пройшли:

- медичний попередній огляд та визнані придатними виконувати електромонтажні роботи на висоті;
- навчання та перевірку знань з електробезпеки;
- навчання в закладах освіти для виконання робіт з підвищеною небезпекою (у професійно-технічних училищах, навчально-курсівих комбінатах, центрах підготовки та перепідготовки робітничих кадрів, в організаціях) за затвердженою програмою;
- спеціальне навчання та атестацію з питань пожежної безпеки;
- вступний інструктаж у службі охорони праці;
- первинний інструктаж безпосередньо на робочому місці.

Робітники повинні бути проінструктовані щодо розпорядку на робочому місці, порядку переміщення по території об'єкта, місце відпочинку під час технологічних та обідньої перерв, порядок закінчення роботи.

До початку робіт у комплексній бригаді проводиться первинний інструктаж з безпечного виконання робіт з основної та суміжних професій та ознайомлення з правилами надання першої допомоги.

Допущені мають виконувати тільки ті роботи, про безпечне виконання яких вони проінструктовані безпосередньо керівником.

Особи з простудними і хронічними захворюваннями верхніх дихальних шляхів до роботи з монтажу електромереж та заготовки і монтажу пластмасових труб не допускаються.

Роботи на висоті (при підйомі над поверхнею вище, ніж 1,3 м) виконуються тільки з риштувань або помостів.

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

5.2 Вимоги безпеки перед початком роботи

До початку робіт керівник зобов'язаний:

- перевірити ступінь готовності будівельних робіт;
- оцінити виробничі обставини, можливість взаємодії з іншими будівельно-монтажними організаціями у відповідності з проектом виконання робіт (ПВР); можливість безпечного застосування машин, механізмів, пристосувань, місця їх установки та порядок проїзду; можливість безпечного застосування піротехнічного інструменту, безпечної подачі електричних конструкцій, електротехнічних апаратів та інших блоків;
- узгодити з відповідними службами та, при необхідності, внести уточнення в ПВР.
- ознайомити працюючих з ПВР та технологічними картами на всі види робіт.

Керівник робіт повинен здійснити первинний інструктаж, який стосується:

- характеру та безпечних методів виконання робіт (у т.ч. за складних погодних умов); порядку проходів до кожного робочого місця;
- наявності небезпечних зон та відкритих каналів і траншей, відкритих прорізів, отворів у перекриттях та стінах;
- порядку розвантаження та складування матеріалів, устаткування та конструкцій;
- місць та порядку підключення зварювальних трансформаторів, трансформаторів безпеки, електрифікованого інструменту, засобів електроосвітлення;
- порядку і місця установки вантажних лебідок та інших механізмів у монтажній зоні; риштувань драбин; наявності діючих електроустановок;
- надання першої допомоги, виклику швидкої медичної допомоги, пожежної охорони, керівника робіт чи роботодавця, представника служби охорони праці.

					ДП.123.211.014.00 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

Перевірити наявність та термін дії посвідчень з охорони праці, електропожежобезпеки, посвідчень на право виконання спеціальних видів робіт (зварювання, монтаж кабельної системи).

Видати наряд-допуск операторам на виконання робіт підвищеної небезпеки з проведенням цільового інструктажу та записом до журналу реєстрації інструктажів з питань охорони праці.

Попередити працюючих, про з'єднання та від'єднання від мережі обладнання, механізмів, інструменту, інвентарних шаф тощо.

Електромонтажнику необхідно отримати перевірені:

- засоби індивідуального захисту;
- захисну каску з підшоломником;
- окуляри захисні;
- спецодяг згідно з існуючими нормами та колективною угодою;
- інструмент індивідуальний:
- ніж монтерський;
- метр складальний або рулетку;
- драбину спеціальну для виконання монтажних робіт;

5.3 Вимоги безпеки під час виконання роботи

Прокладання кабелів слід виконувати тільки в рукавицях.

Працювати ручними ударними інструментами слід із застосуванням захисних щитків або окулярів з непробивним склом.

Переносити чи перевозити інструмент з гострими кутами треба лише в чохлах.

Не дозволяється розміщувати кабель, барабан з кабелем та без нього, механізми, пристрої та інструменти безпосередньо біля бровки траншеї.

Переміщувати барабан з кабелем слід після того, коли цвяхи із щік барабана вийнято, а кінці кабелю надійно закріплено.

					ДП.123.211.014.00 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

Перекочувати барабан з кабелем слід у напрямку стрілки, нанесеної фарбою на щоці барабана.

Переміщувати барабан з кабелем вручну дозволяється тільки по твердому ґрунту або надійному настилу по горизонтальній поверхні на відстань не більше.

Не дозволяється працюючим чи стороннім особам перебувати на шляху барабана, що переміщується.

Під час піднімання барабана необхідно слідкувати за тим, щоб не пошкодити щоки барабана та втулку.

Перед розмотуванням барабан встановити на домкрати (чи інший підймальний пристрій). Підняти його на 0,15-0,20 м над поверхнею землі, кузова автомобіля тощо, щоб барабан міг вільно обертатися та не зміщуватися по осі.

Барабан встановити так, щоб кабель розмотувався з його верхньої частини.

Розмотувати кабель з барабана слід тільки за наявності гальмівного пристрою.

На трасах з поворотами робітники не повинні перебувати всередині кута повороту кабелю, а також підтримувати або відтягувати кабель руками. Для цього на поворотах повинні бути встановлені кутові ролики.

За умов прокладання кабелю вручну всі робітники повинні перебувати з одного боку кабелю.

При розмотуванні кабелю за допомогою привідної лебідки робітникам, який знаходиться біля неї, необхідно слідкувати за роботою лебідки і контролювати зусилля натягування. Якщо розмотування виконується ручною лебідкою, то крутити її та контролювати зусилля повинні два робітники.

Розміщення конструкцій і пристроїв для витягнення кабелю з барабана повинно бути таким, щоб забезпечити прохід робітника, який супроводжує кінець кабелю з тросом, без перепон і торкання раніше прокладених кабелів.

Роботу виконувати тільки за умов наявності інвентарних засобів підмащування, які пройшли періодичне випробування.

					ДП.123.211.014.00 ПЗ	Арк.
						61
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

Протягувати кабель через отвори в стінах та через міжповерхові перекриття дозволяється за умов, коли робітники знаходяться по обидва боки.

Тягнути кабель по трасі слід тільки за умов надійного візуального, телефонного чи радіозв'язку з керівником робіт.

Не дозволяється перекладати кабель, який знаходиться під напругою.

Під час виконання монтажних робіт на перетинах з автошляхами та інженерними мережами умови виконання робіт слід узгоджувати з їх власниками.

5.4 Вимоги безпеки після закінчення роботи

Не залишати робоче місце до закріплення кабелю на кабельних конструкціях.

Упорядкувати робоче місце.

Прибрати інструмент та пристрої у відведене для них місце.

Зняти спецодяг, засоби індивідуального захисту, очистити від пилу, скласти у відведене для них місце, помити руки, обличчя з милом; при можливості, прийняти душ.

Доповісти керівникові робіт про всі недоліки, які мали місце під час виконання робіт.

5.5 Вимоги безпеки в аварійних ситуаціях

У разі виникнення аварійної ситуації (поява диму, запаху газу, обвалення кабельних споруд тощо) слід припинити роботу.

Огородити небезпечну зону, залишити її, не допускати в неї сторонніх осіб.

Повідомити про аварійну ситуацію або нещасний випадок керівника робіт.

Якщо є потерпілі, надати їм першу медичну допомогу; при необхідності, викликати швидку медичну допомогу.

					ДП.123.211.014.00 ПЗ	Арк.
						62
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

5.6 Перша допомога при нещасних випадках

При ураженні електричним струмом необхідно негайно звільнити потерпілого від дії електричного струму, відключивши електроустановку від джерела живлення, а при неможливості відключення -відтягнути його від струмоведучих частин за одяг або застосувавши підручний ізоляційний матеріал.

При відсутності у потерпілого дихання і пульсу необхідно робити йому штучне дихання і непрямий (зовнішній) масаж серця, звернувши увагу на зіниці. Розширені зіниці свідчать про різке погіршення кровообігу мозку. При такому стані оживлення починати необхідно негайно, після чого викликати швидку медичну допомогу.

Для надання першої допомоги при пораненні необхідно розкрити індивідуальний пакет, накласти стерильний перев'язочний матеріал, що міститься у ньому, на рану і зав'язати її бинтом.

Якщо індивідуального пакету якимсь чином не буде, то для перев'язки необхідно використати чисту носову хустинку, чисту полотняну ганчірку і т. ін. На те місце ганчірки, що приходить безпосередньо на рану, бажано накапати декілька крапель настойки йоду, щоб одержати пляму розміром більше рани, а після цього накласти ганчірку на рану. Особливо важливо застосовувати настойку йоду зазначеним чином при забруднених ранах.

При переломах і вивихах кінцівок необхідно пошкоджену кінцівку укріпити шиною, фанерною пластинкою, палицею, картоном або іншим подібним предметом. Пошкоджену руку можна також підвісити за допомогою перев'язки або хустки до шиї і прибинтувати до тулуба.

При переломі черепа (несвідомий стан після удару голови, кровотеча з вух або роту) необхідно прикласти до голови холодний предмет (грілку з льодом, снігом чи холодною водою) або зробити холодну примочку.

При підозрінні перелому хребта необхідно потерпілого покласти на дошку, не підіймаючи його, чи повернути потерпілого на живіт обличчям униз,

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

наглядаючи при цьому, щоб тулуб не перегинався, з метою уникнення ушкодження спинного мозку.

При переломі ребер, ознакою якого є біль при диханні, кашлю, чханні, рухах, необхідно туго забинтувати груди чи стягнути їх рушником під час видиху.

При опіках вогнем, парою, гарячими предметами ні в якому разі не можна відкривати пухирі, які утворюються, та перев'язувати опіки бинтом.

При опіках першого ступеня (почервоніння) обпечене місце обробляють ватою, змоченою етиловим спиртом.

При опіках другого ступеня (пухирі) обпечене місце обробляють спиртом або 3%-ним марганцевим розчином.

При опіках третього ступеня (зруйнування шкіряної тканини) накривають рану стерильною пов'язкою та викликають лікаря.

Перша допомога при кровотечі: підняти поранену кінцівку вгору. Рану закрити перев'язочним матеріалом (із пакета), складеним у клубочок, придавити її зверху, не торкаючись самої рани, потримати на протязі 4-5 хвилин. Якщо кровотеча зупинилася, не знімаючи накладеного матеріалу, поверх нього покласти ще одну подушечку з іншого пакета чи кусок вати і забинтувати поранене місце (з деяким натиском).

У разі сильної кровотечі, яку не можна зупинити пов'язкою, застосовується здавлювання кровоносних судин, які живлять поранену область, за допомогою згинання кінцівок в суглобах, а також пальцями, джгутом або закруткою, У разі великої кровотечі необхідно терміново викликати лікаря.

Якщо сталася пожежа, необхідно викликати пожежну частину і приступити до її гасіння наявними засобами пожежогасіння.

В усіх випадках необхідно виконувати вказівки керівника робіт по ліквідації наслідків аварії.

					ДП.123.211.014.00 ПЗ	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Розробка симулятора BIOS мала на меті створення онлайн-середовища, яке дозволяє студентам та учням старших класів безпечно знайомитися з можливостями BIOS. Основний акцент було зроблено на інтерактивності, зручності у використанні та максимально точному відтворенні реального інтерфейсу BIOS.

У ході реалізації проєкту було створено систему управління за допомогою клавіатури, систему паролів для обмеження доступу, можливість збереження та відновлення налаштувань через LocalStorage, а також систему завдань для кращого засвоєння матеріалу. Навігація в симуляторі повністю відповідає реальному BIOS: користувач може перемикатися між вкладками, змінювати параметри, зберігати або скасовувати зміни. Візуальне оформлення також максимально наближене до реального BIOS, що забезпечує користувачеві відчуття роботи зі справжнім інтерфейсом.

Робота над проєктом дозволила освоїти JavaScript на новому рівні, а також вивчити технології, з якими раніше не було досвіду, зокрема LocalStorage. Це дало змогу реалізувати збереження та відновлення користувацьких налаштувань без використання серверної частини, що підвищує зручність роботи з симулятором.

Симулятор BIOS повністю інтерактивний і відповідає всім початковим вимогам. Він може мати практичну цінність для навчальних закладів або компаній, що займаються розробкою та тестуванням комп'ютерного обладнання. Функціональність проєкту реалізована в повному обсязі, і немає необхідності у його доопрацюванні.

Під час розробки не виникало значних труднощів, оскільки всі функції вдалося реалізувати відповідно до початкового задуму. Основний висновок після виконання роботи — будь-який складний задум можна реалізувати, якщо приділяти достатньо часу навчанню та вдосконаленню навичок.

					ДП.123.211.014.00 ПЗ	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. CSS-хитрощі. Як нові властивості змінюють розробку URL: <https://dou.ua/forums/topic/51341/> (дата звернення: 28.11.2024).
2. Flexbox у CSS & HTML (Посібник) – адаптивні макети URL: <https://shorturl.at/TsPJw> (дата звернення: 11.11.2021).
3. HTML and CSS (DOU) URL: <https://quizlet.com/pt/867960199/html-and-css-dou-flash-cards/> (дата звернення: 05.07.2023).
4. Адаптивна верстка: що вона охоплює та яке значення має URL: <https://atlasiko.com.ua/blog/web-development/adaptive-layout/> (дата звернення: 12.07.2021).
5. Адаптивний дизайн: інновації та перспективи веб-розробки URL: <https://bdut.co.ua/pro-nas/adaptyvnyy-dyzayn/> (дата звернення: 12.11.2021).
6. Веб-програмування та веб-дизайн URL: <https://shorturl.at/yeior> (дата звернення: 14.01.2022).
7. Використання медіа-запитів та гнучких макетів для адаптивного дизайну URL: <https://dou.ua/forums/topic/51341/> (дата звернення: 12.09.2024).
8. Зберігання даних у браузері за допомогою LocalStorage URL: <https://uk.javascript.info/localstorage> (дата звернення: 12.05.2024).
9. Калініченко А. О. «Організація дипломного проектування фахового молодшого бакалавра». Методичні вказівки для виконання студентами спеціальності 123 «Комп'ютерна інженерія» – Ніжин: , 2024. – 80 с.
10. Ключові інструменти та технології Front-end розробки веб-сайтів URL: <https://webbookstudio.com/ua/articles/key-tools-and-technologies-used-in-front-end-website-development/> (дата звернення: 07.01.2025).
11. Основні тренди у веб-дизайні URL: <https://shorturl.at/RZlar> (дата звернення: 11.11.2021).

					ДП.123.211.014.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		66

12. Підтримка новіших JS функцій в старих браузерх за допомогою поліфілів та транспіляції URL: <https://shorturl.at/WExFo> (дата звернення: 22.12.2021).

13. Реалізація бізнес-логіки на JavaScript у фронтенд-розробці URL: <https://dou.ua/forums/topic/33590/> (дата звернення: 15.12.2021).

14. Співбесіда фронтенд URL: <https://shorturl.at/UNnPf> (дата звернення: 14.10.2023).

15. Як стати веб-розробником з нуля: кроки та план навчання URL: <https://foxminded.ua/yak-staty-web-rozrobnykom-z-nulia/> (дата звернення: 04.09.2023).

					ДП.123.211.014.00 ПЗ	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		67