

## ЗМІСТ

<b>ВСТУП.....</b>	<b>3</b>
<b>1. ОГЛЯД ПІДПРИЄМСТВА .....</b>	<b>5</b>
<b>1.1. СТРУКТУРА ТА НАПРЯМКИ ДІЯЛЬНОСТІ ПІДПРИЄМСТВА .....</b>	<b>5</b>
1.1.1. Загальна інформація про підприємство .....	5
1.1.2. Опис установи .....	5
<b>1.2. КОМП'ЮТЕРИЗОВАНІ ЗАСОБИ ПІДТРИМКИ ДІЯЛЬНОСТІ ПІДПРИЄМСТВА .....</b>	<b>5</b>
1.2.1. Робоче місце при проходженні практики .....	5
1.2.2. Опис комп'ютерної техніки і різновидів використовуваного програмного забезпечення .....	5
<b>2. ВИКОНАННЯ ЗАВДАННЯ ПРАКТИКИ .....</b>	<b>6</b>
<b>2.1. НАЛАШТУВАННЯ РОБОЧОГО СЕРЕДОВИЩА.....</b>	<b>6</b>
<b>2.2. ЗМІСТ ЗАВДАННЯ.....</b>	<b>6</b>
<b>2.3. ВИКОНАННЯ ПОСТАВЛЕНОГО ЗАВДАННЯ .....</b>	<b>7</b>
2.3.1. Сторінка Login .....	7
2.3.2. Сторінка Home .....	8
2.3.3. Сторінка Surveys list.....	9
2.3.4. Сторінка New survey.....	11
2.3.5. Сторінка проходження тестування.....	12
<b>3. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ ВІД УНІВЕРСИТЕТУ .....</b>	<b>14</b>
<b>3.1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....</b>	<b>14</b>
3.1.1. Мова програмування .....	14
3.1.2. Кар'єрний ріст розробника.....	15
3.1.3. Тести.....	16
3.1.4. Аналіз потреб користувача .....	17
<b>3.2. ДОСЛІДЖЕННЯ АНАЛОГІВ.....</b>	<b>18</b>
3.2.1. Опис та аналіз аналогів.....	18
3.2.2. Індивідуальне завдання.....	19
<b>ВИСНОВОК .....</b>	<b>21</b>

## ВСТУП

Метою наукової практики є застосування студентом набутих навичок на підприємстві та здобуття студентом професійних навичок і вмінь здійснення самостійної наукової діяльності, а також написання опису предметної області та аналогів дипломної роботи. Вона спрямована на закріплення теоретичних знань, одержаних студентами під час навчання, набуття і удосконалення практичних навичок і умінь у процесі професійної діяльності, розвиток у студентів здатності компетентного прийняття рішень у виробничих ситуаціях, оволодіння сучасними методами та формами наукової діяльності.

Керівник підприємства, після ознайомлення студента зі структурою закладу та процесом роботи, дає індивідуальне завдання, яке студент має виконати в межах даної практики. Студент також повинен виділити час для написання опису предметної області та аналогів дипломної роботи.

Практика проходила під керівництвом штатного працівника фірми. Керівником практики було поставлене наступне завдання для студента: спроектувати, реалізувати програмними засобами та протестувати веб-додаток. Темою програмного продукту було «Система стандартизованого тестування студентів».

Стандартизований тест або іспит являє собою тестування, яке проводиться та перевіряється заздалегідь установленим, або «стандартним», шляхом. Стандартизовані тести розроблені таким чином, що питання, умови проведення, оцінювання і тлумачення заздалегідь погоджені, процеси проведення і оцінювання визначені наперед стандартним чином. Стандартне тестування називають найбільш значним соціальним внеском сучасної психології.

Будь-який тест, в якому той же самий тест дається таким же чином для всіх тестованих, являє собою стандартизований тест. Стандартизовані тести — це не обов'язково тести з високими ставками, тести з обмеженим часом, або тести з декількома варіантами відповідей. Протилежністю стандартизованого тестування є нестандартизоване тестування, в якому або істотно різні випробування наведені до різних тестованих, або той же самий тест

призначається при істотно різних умовах (наприклад, одній групі дозволено набагато менше часу, щоб завершити випробування, ніж наступній групі) або оцінюватися по-різному (наприклад, та же відповідь вважається правильним для одного студента, але неправильно для іншого студента).

Стандартизовані тести сприймаються як більш справедливі, ніж не-стандартизовані тести, тому що кожен отримує той же самий тест і ту ж систему класифікації. Це є більш справедливим і більш об'єктивним, ніж система, в якій деякі студенти отримують більш простий тест і інші отримують більш важке випробування. Узгодженість також дозволяє більш надійне порівняння результатів у всіх тестованих, тому що кожен приймає один і той же тест.

**Актуальність** даної теми полягає у тому, що навчальним закладам потрібно перевіряти рівень знань студентів, які в них навчаються, не витрачаючи багато часу на різні багато часові опитування.

**Метою** є розробка web-додатку, яким будуть користуватись навчальні заклади. Вони зможуть з легкістю додати в додаток свої бібліотеки питань та відповідей до них, та використовувати їх для тестування різного рівня студентів. Як результат кожен додаток буде мати свою базу даних, та свої унікальні питання, тому що як ми знаємо, рівень оцінювання студентів різних курсів у кожного свій.

# **1. ОГЛЯД ПІДПРИЄМСТВА**

## **1.1. СТРУКТУРА ТА НАПРЯМКИ ДІЯЛЬНОСТІ ПІДПРИЄМСТВА**

### **1.1.1. Загальна інформація про підприємство**

ФОП Запорожець М. В. – підприємство, яке займається розробкою програмних продуктів, які являють собою веб-додатки, на замовлення іноземних замовників. Діяльність підприємства тісно пов'язана з веб-програмуванням, то му завданням практики було вибрано саме створення веб-додатку.

### **1.1.2. Опис установи**

Підприємство ФОП Запорожець М. В. було створене у 2012 році. Офіс компанії розташований у невеликому районному центрі Черкас. В орендованому офісі працюють кілька співробітників, включно із директором та власником підприємства. Більшість працівників, інженерів-розробників програмного забезпечення, працюють віддалено через мережу інтернет. Такий підхід дозволяє динамічно регулювати кількість працівників на проекти різного масштабу та складності.

## **1.2. КОМП'ЮТЕРИЗОВАНІ ЗАСОБИ ПІДТРИМКИ ДІЯЛЬНОСТІ ПІДПРИЄМСТВА**

### **1.2.1. Робоче місце при проходженні практики**

Робоче місце складалося з комп'ютерного столу, ноутбука ASUS X552MD, мишки. В наявності було також високошвидкісне підключення до інтернету.

### **1.2.2. Опис комп'ютерної техніки і різновидів використовуваного програмного забезпечення**

На практиці використовувалась така комп'ютерна техніка: ноутбук із характеристиками, представленими в табл. 1:

На робочій станції встановлена ліцензійна операційна система Microsoft Windows 10.

<b>Процесор</b>	Intel(R) Celeron(R) CPU N2830 @ 2.16GHz
<b>Материнська плата</b>	ASUS X550MD
<b>Оперативна пам'ять</b>	4GB
<b>Відеокарти</b>	Intel Bay Trail
	nVIDIA GeForce 820M (Asus)
<b>Жорсткий диск</b>	500 GB TOSHIBA MQ01ABF050

Таблиця 1 – Характеристики комп'ютера

## **2. ВИКОНАННЯ ЗАВДАННЯ ПРАКТИКИ**

### **2.1. НАЛАШТУВАННЯ РОБОЧОГО СЕРЕДОВИЩА**

Одним із завдань практики було встановити на робочу станцію таке програмне забезпечення:

- Web Storm 2020.1 (для розробки продукту на мовах HTML, CSS, TypeScript);
- Angular CLI v9.2.4 (для тестування web-додатку в процесі розробки);
- Fork v1.45.2 (для завантаження файлів проекту в систему контролю версій GitHub).

### **2.2. ЗМІСТ ЗАВДАННЯ**

Керівником практики було поставлено завдання розробити web-додаток на тему «Система стандартизованого тестування студентів». Додаток має містити базу даних студентів та тестових завдань і відповідей до них. Також система

передбачає наявність питань з розгорнутою відповіддю на кілька речень.

Доступ до створення нового тесту, а також його перевірки та повідомлення результатів студентів має лише адміністратор.

## **2.3. ВИКОНАННЯ ПОСТАВЛЕНОГО ЗАВДАННЯ**

Для реалізації завдання була обрана IDE Web Storm. Розробка виконувалася у складі 1 студента та штатного front-end розробника. Декомпозицію проекту на окремі завдання та їхній розподіл виконувалося team-leader. Кожен модуль містив створення інтерфейсу користувача, а створення взаємодії із сервером виконувалося іншим членом команди.

Модуль, що розроблявся мною, відповідав за інтерфейс для користувачів та адміністратора.

База даних для проекту проектувалася та back-end програмного продукту розроблявся іншим членом команди.

Весь проект було виконано з допомогою фреймворку Angular та на основі програмного модульного комплексу Node.js.

У результаті виконання завдання було отримано програмний компонент, який дозволяє поводити стандартизоване тестування студентів для виявлення рівня набутих ними знань у процесі навчання.

### **2.3.1. Сторінка Login**

Сторінка Login являє собою сторінку входу до системи. На ній реалізовано 2 поля – Login та Password – за допомогою них здійснюється вхід в систему. Для валідації даних на стороні користувача було застосовано стандартний функціонал валідації реактивних форм фреймворку Angular.

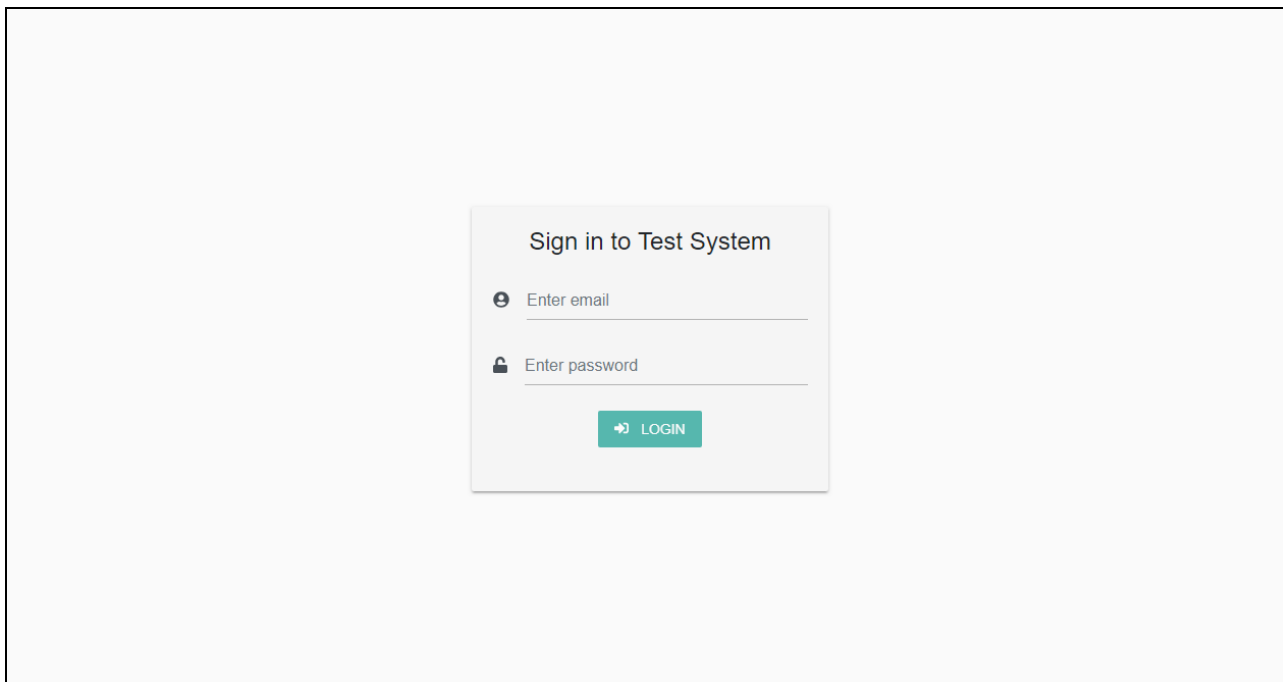


Рисунок 1 – Сторінка Login.

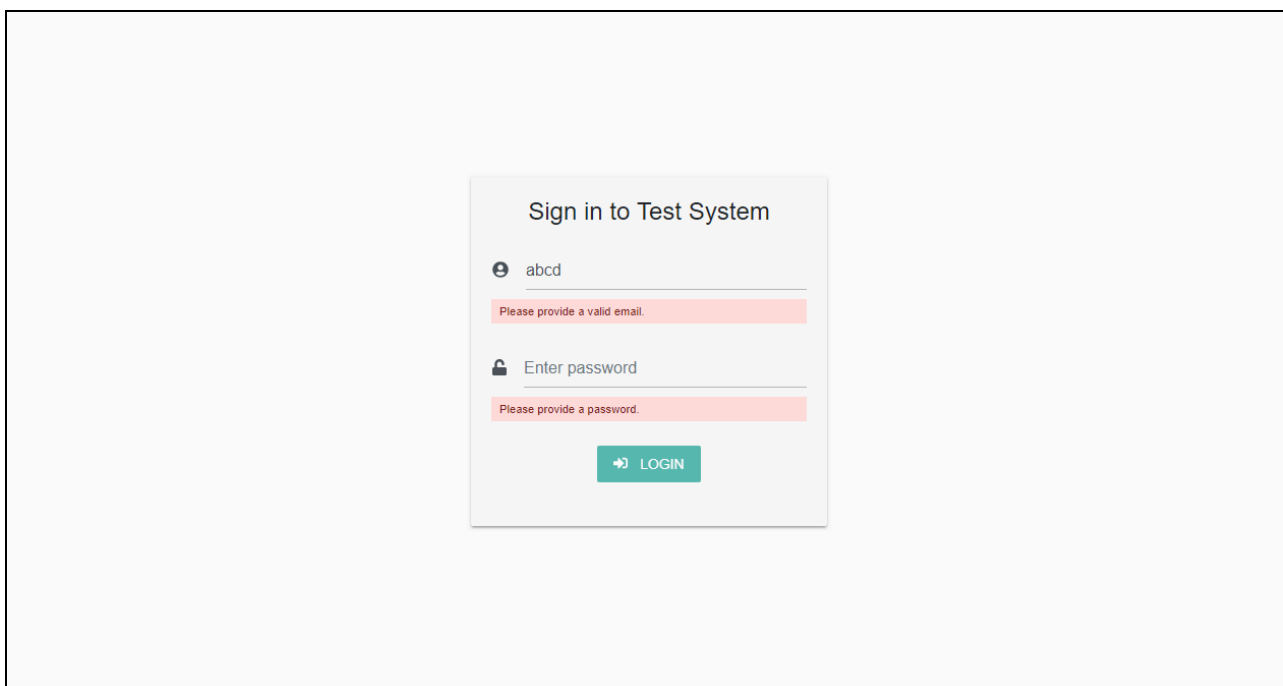


Рисунок 2 – Сторінка Login. Демонстрація валідації.

### 2.3.2. Сторінка Home

На сторінці Home виводиться основна статистична інформація програмного продукту, а саме кількість користувачів та кількість різних згенерованих тестів розбитих на категорії по статусу проходження тестування.

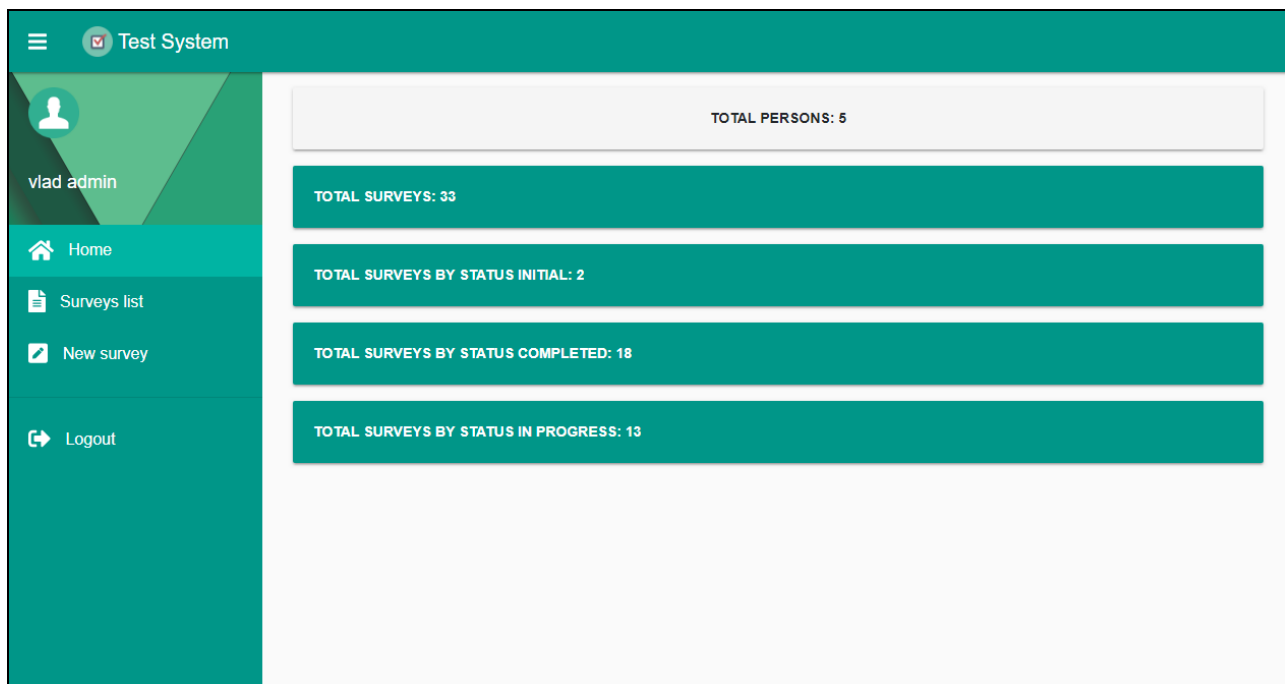
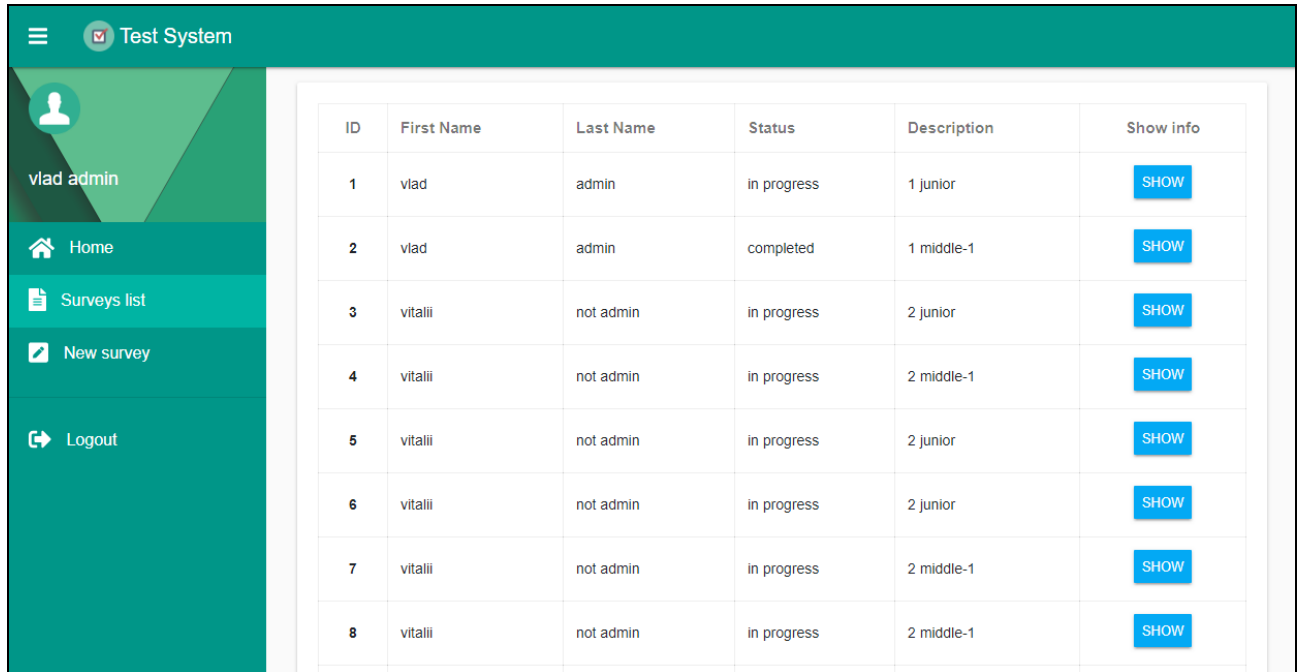


Рисунок 1 – Загальний вигляд сторінки Home.

### 2.3.3. Сторінка Surveys list

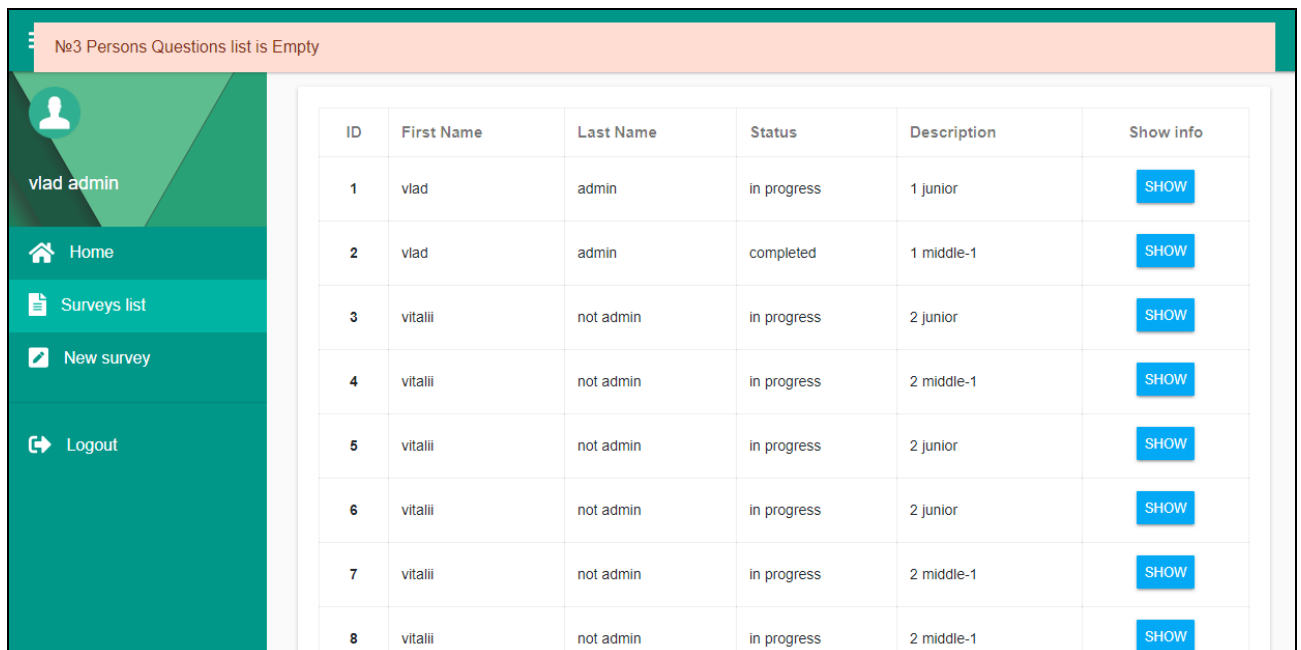
Сторінка Surveys list дозволяє переглянути всі згенеровані тести та студенти, яким вони присвоєні, а також їх статус та можливість перегляду результатів кожного пройденого тесту. Виведення таблиці забезпечено можливостями CSS-бібліотеки Bootstrap Material Design. При натисненні на кнопку “SHOW” відкривається нова сторінка із результатами пройденого тесту. Якщо ж тест ще не пройдено, то буде виведено помилку про те, що перегляд результатів на даний момент не можливий.





ID	First Name	Last Name	Status	Description	Show info
1	vlad	admin	in progress	1 junior	<a href="#">SHOW</a>
2	vlad	admin	completed	1 middle-1	<a href="#">SHOW</a>
3	vitalii	not admin	in progress	2 junior	<a href="#">SHOW</a>
4	vitalii	not admin	in progress	2 middle-1	<a href="#">SHOW</a>
5	vitalii	not admin	in progress	2 junior	<a href="#">SHOW</a>
6	vitalii	not admin	in progress	2 junior	<a href="#">SHOW</a>
7	vitalii	not admin	in progress	2 middle-1	<a href="#">SHOW</a>
8	vitalii	not admin	in progress	2 middle-1	<a href="#">SHOW</a>

Рисунок 1 – Загальний вигляд сторінки Surveys list.



№3 Persons Questions list is Empty

ID	First Name	Last Name	Status	Description	Show info
1	vlad	admin	in progress	1 junior	<a href="#">SHOW</a>
2	vlad	admin	completed	1 middle-1	<a href="#">SHOW</a>
3	vitalii	not admin	in progress	2 junior	<a href="#">SHOW</a>
4	vitalii	not admin	in progress	2 middle-1	<a href="#">SHOW</a>
5	vitalii	not admin	in progress	2 junior	<a href="#">SHOW</a>
6	vitalii	not admin	in progress	2 junior	<a href="#">SHOW</a>
7	vitalii	not admin	in progress	2 middle-1	<a href="#">SHOW</a>
8	vitalii	not admin	in progress	2 middle-1	<a href="#">SHOW</a>

Рисунок 2 – Демонстрація виведення помилки про неможливість перегляду результатів тесту в даний момент.

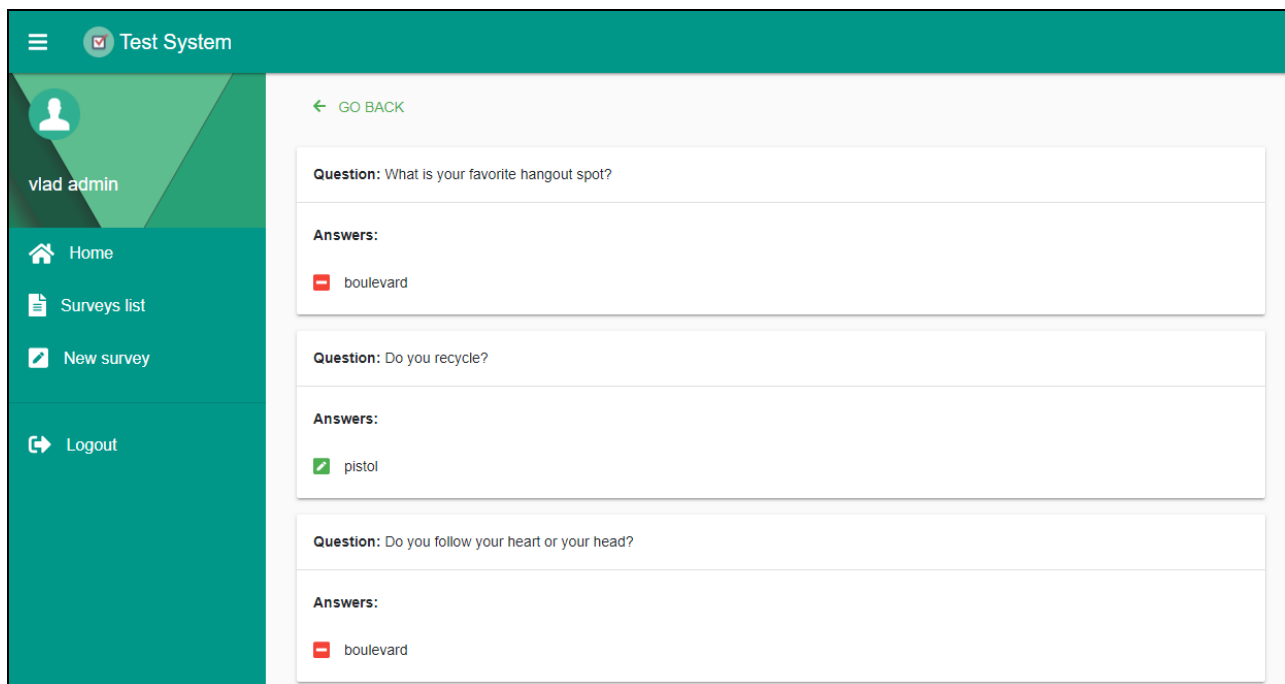


Рисунок 3 – Перегляд результатів вже пройденого тесту.

### 2.3.4. Сторінка New survey

Сторінка New survey дозволяє відправити команду на генерацію нового тесту для студента із заданим рівнем складності. Пошук студента по імені реалізований за схемою Ternhead. Тобто при введенні в поле автоматично запускається пошук та виводяться знайдені студенти. Після відправлення команди, з даних що приходять від сервера, формується посилання для незареєстрованих користувачів на проходження даного тесту.

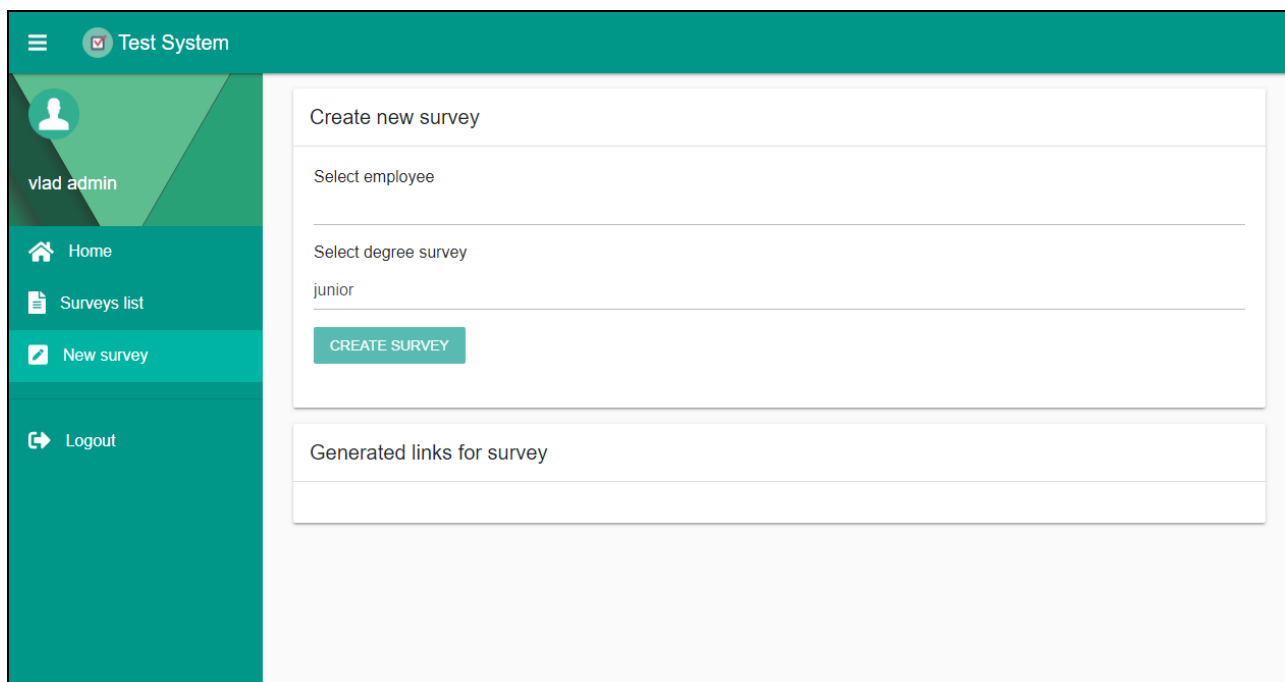


Рисунок 1 – Загальний вигляд сторінки New survey.

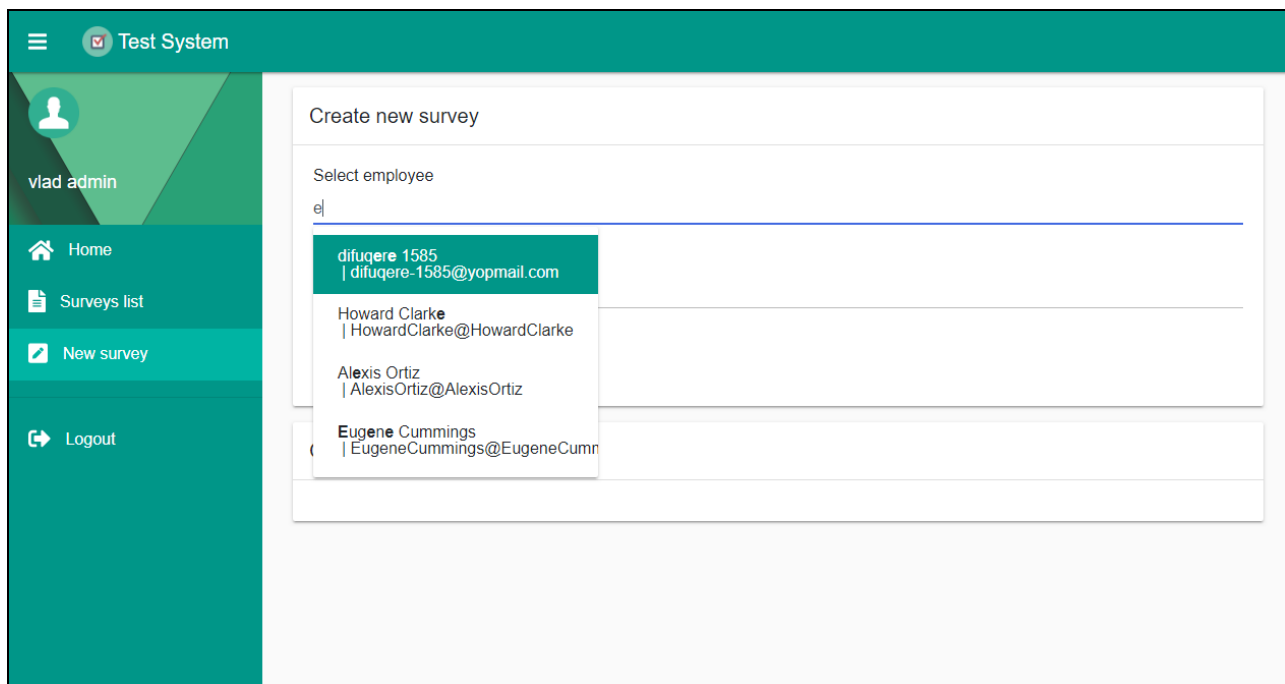


Рисунок 2 – Демонстрація пошуку студента, реалізованого за схемою Теруhead.

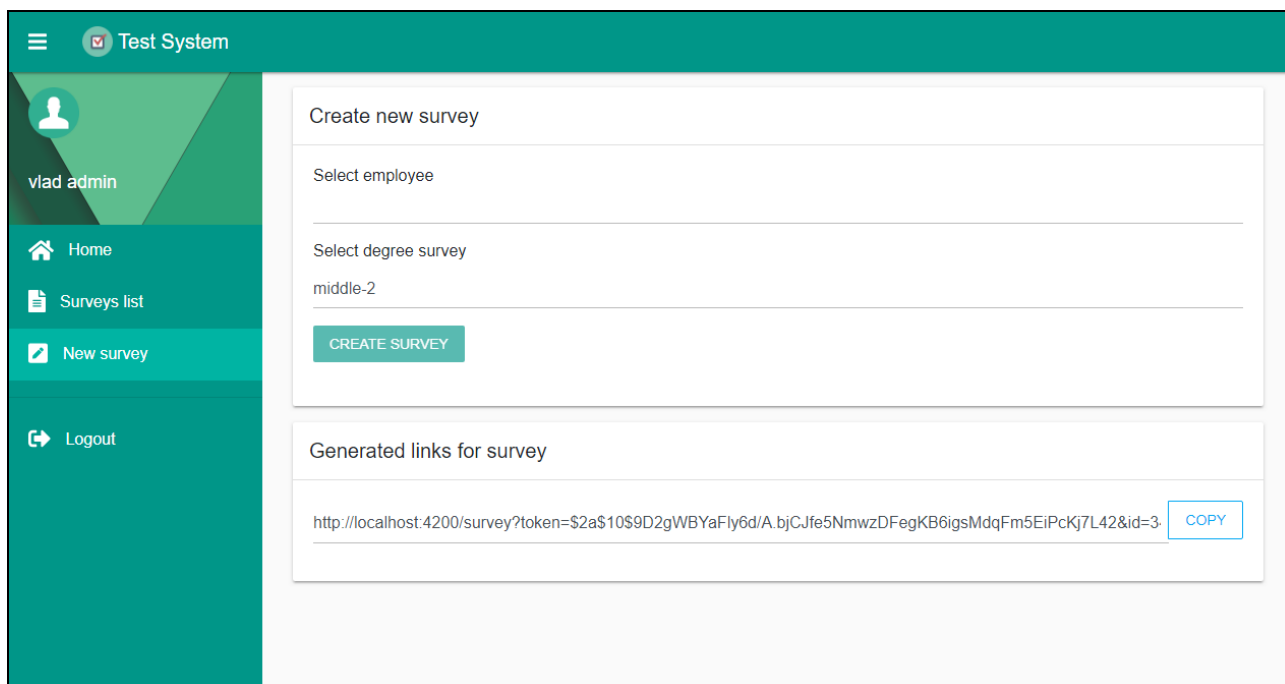
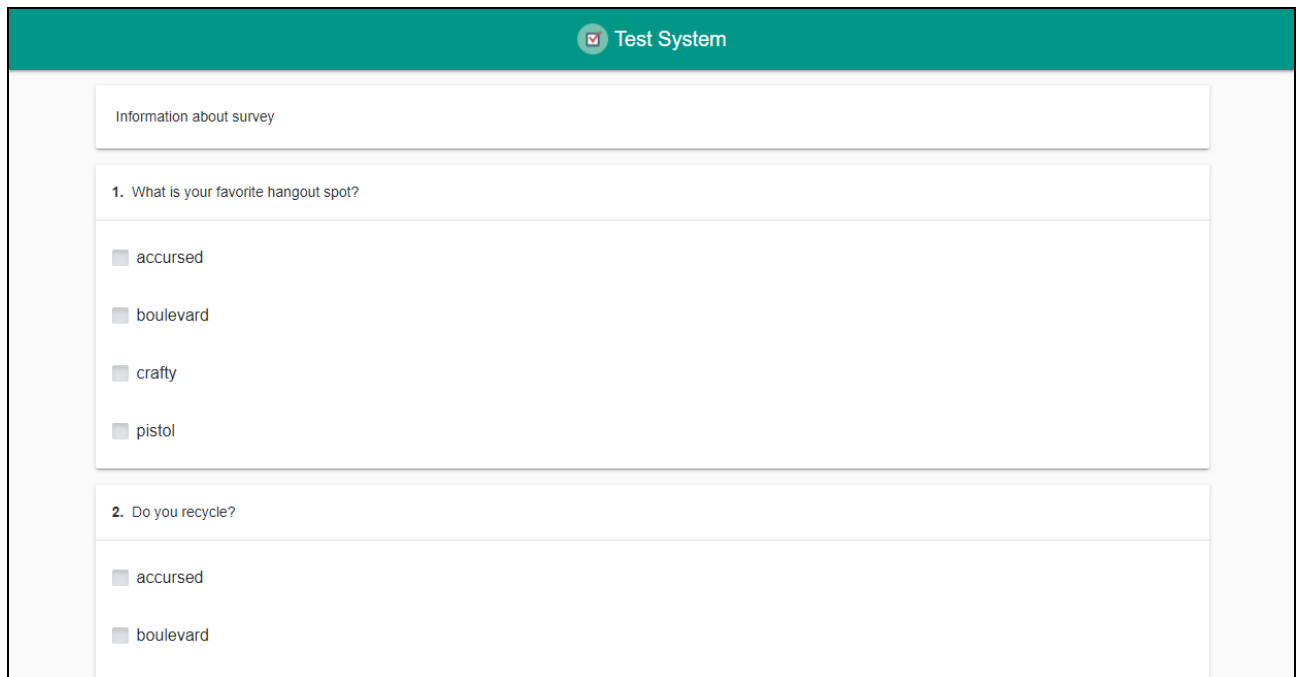


Рисунок 3 – Демонстрація згенерованого посилання, готового для копіювання, для проходження тестування незареєстрованим користувачем.

### 2.3.5. Сторінка проходження тестування

Сторінка проходження тестування дозволяє вибрати відповіді на тестові питання та ввести розгорнуту відповідь на питання, які це вимагають. Після введення всіх відповідей користувач натискає кнопку «Send». Після цього дані відправляються на сервер для опрацювання на визначення правильності/не

правильності вибраних/введених відповідей.



The image shows a web-based test system interface. At the top, there is a teal header bar with a circular icon containing a checkmark and the text "Test System". Below the header, the main content area is white and contains a survey. The survey starts with a section titled "Information about survey". The first question is "1. What is your favorite hangout spot?". It has four radio button options: "accursed", "boulevard", "crafty", and "pistol". The second question is "2. Do you recycle?". It has two radio button options: "accursed" and "boulevard".

Test System

Information about survey

1. What is your favorite hangout spot?

☐ accursed

☐ boulevard

☐ crafty

☐ pistol

2. Do you recycle?

☐ accursed

☐ boulevard

Рисунок 1 – Загальний вигляд сторінки проходження тестування.

### 3. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ ВІД УНІВЕРСИТЕТУ

#### 3.1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

##### 3.1.1. Мова програмування

**Мова програмування** – це штучна мова, призначена для запису комп'ютерних програм. Мови програмування класифікують за такими основними критеріями.

##### **Рівень абстракції.**

- 1) Мови програмування низького рівня (машинно-орієнтовані).
- 2) Мови програмування високого рівня.
- 3) Область застосування.
- 4) Універсальні.
- 5) Спеціалізовані.
- 6) Підтримувані парадигми програмування.
- 7) Об'єктно-орієнтовані.
- 8) Логічні.
- 9) Функціональні, тощо.

**Мови програмування низького рівня** (машинно-орієнтовані) - такі, у яких принципи керування і структура даних безпосередньо відображають архітектуру ЕОМ. Тобто, такі мови орієнтовані на конкретний тип процесора і враховують його особливості.

**Мови програмування високого рівня** - такі, у яких засоби керування і роботи з даними відображають потреби програміста, а не можливості апаратних засобів. Програми, що складаються на таких мовах, являють собою послідовності операторів, структуровані відповідно до правил мови. Вони оперують сутностями, ближчими і зрозумілішими людині, такими як змінні, функції тощо. Особливості конкретних комп'ютерних архітектур в них не

враховуються, тому розроблені на цих мовах програми можуть виконуватися на інших комп'ютерах тієї ж платформи.

Основні відмінності мови програмування високого рівня від машинно-орієнтованої :

1) алфавіт алгоритмічної мови значно ширше алфавіту машинно-орієнтованої, що істотно підвищує наочність тексту такої програми;

2) набір допустимих в алгоритмічній мові операцій не залежить від набору машинних операцій, а вибирається із міркувань зручності формулювання послідовності дій;

3) формат речень у алгоритмічній мові досить гнучкий, що дозволяє за допомогою одного речення задати досить змістовний етап обробки даних;

4) необхідні операції в алгоритмічній мові задаються в зручному для людини вигляді, наприклад, за допомогою загальноприйнятих математичних позначень;

5) в алгоритмічній мові може бути передбачений значно більший набір типів даних у порівнянні з набором машинних типів даних.

### **3.1.2. Кар'єрний ріст розробника**

1) Junior (джуніор) - перша сходинка на довгому шляху саморозвитку. Як правило, для успішного проходження цього етапу потрібна допомога більш досвідчених товаришів. Серйозні проектні завдання Джуніору ще не під силу, проте з кожним днем він отримує все більше досвіду і впевненості у власних силах. Основне завдання на старті - це зв'язати великі теоретичні пізнання з реальною практикою промислового програмування. Далі йде рутинний процес роботи, який рано чи пізно призводить до позиції мідла.

2) Middle (мідл) - золота середина, людина, яка вже знає свої можливості. Якщо Джуніор має репутацію оптимістів, то мідл реально оцінюють час і складність поставлених перед ними завдань. Книги вже прочитані, за плечима кілька проектів і можна спокійно працювати і пити каву в перервах. Однак, спокійним і розміреним життя у програмістів бути не може, тому що нові технології розвиваються безперервно і якщо не слідувати за ними по п'ятах,

можна залишитися на незатребуваною узбіччі кар'єри. Мідл, який зрозумів це просте правило, має всі шанси перейти на наступну позицію.

3) Senior (сеньйор) - насправді, це останній етап, що вимагає застосування навичок написання коду в бою. Сеньйор багато часу проводить на мітингах, обговорюючи високорівневі концепції програмного продукту. Найчастіше, саме сеньйори займаються підготовкою джуніор, оскільки у мідлів на це не вистачає часу. Насправді, компанії не проти, якщо людина буде залишатися на цій позиції довго і щасливо. Термін, проведений на цій щаблі кар'єри залежить тільки від бажання програміста.

### **3.1.3. Тести**

Створення тесту – складна і клопітка робота, яка має певний алгоритм. Створення надійного тестового інструменту включає: визначення мети тестування; відбір навчального матеріалу, рівень засвоєння якого необхідно перевірити; вибір форматів тестових завдань; планування часу виконання тесту та темпу виконання окремих завдань; розробка інструктивних матеріалів; обробка та аналіз результатів.

Тест повинен бути достовірним, науковим, несуперечливим.

Вірогідність, науковість, несуперечність тесту – міра істинності тесту, погодженості, відповідності сучасному стану науки й технології, методиці навчання. Необхідно забезпечити погодженість завдань тесту як між собою (внутрішня погодженість), так і з іншими тестами (зовнішня погодженість).

Валідність тесту виступає як найважливіший критерій його доброякісності й визначається як його адекватність і дієвість. У найбільш загальному формулюванні валідність розглядається як характеристика тестів, що вказує на те, що вимірюється і наскільки добре здійснюється це вимірювання. Виходячи з цього, тест вважається валідним, якщо він вимірює те, для чого він призначений.

### 3.1.4. Аналіз потреб користувача

Світова індустрія інформаційних технологій перебуває на швидкості, щоб досягти \$ 5 трлн. У 2019 році, повідомляє дослідницька консультація IDC. Величезна індустрія є функцією багатьох тенденцій, що обговорюються в цьому звіті. Економіка, робота і особисте життя стають все більш цифровими, більш пов'язаними і все більш і більш автоматизованими. Хвилі інновацій будуються з плином часу, приводячи в дію двигун росту технологій, який, здається, знаходиться на порозі ще одного великого стрибка вперед.

За останні два роки кількість спеціалістів в топ-50 найбільших ІТ-компаніях виросла більше ніж на третину — з 43 тис. до 58 тис.

З липня по січень кількість фахівців зросла на 4 731 (11.6 %) в топ-25 та 5 797 (11%) в топ-50 порівняно з першим півріччям 2018-го. Темпи зростання топ-25 стали рекордними за останні п'ять років. (рисунок 1; рисунок 2;)

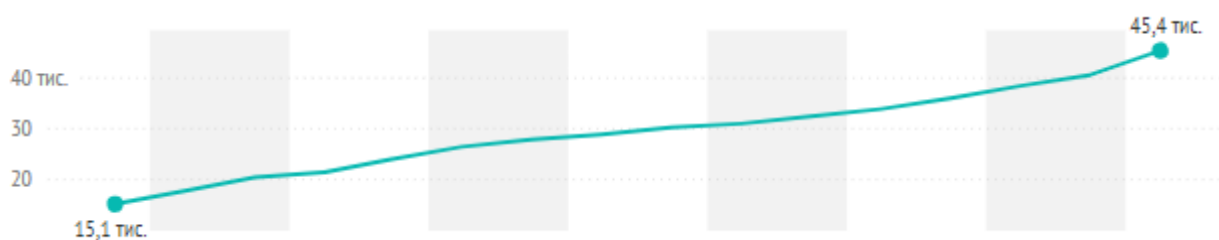


Рисунок 1 – Зростання загальної кількості спеціалістів в 25 найбільших ІТ-компаніях України

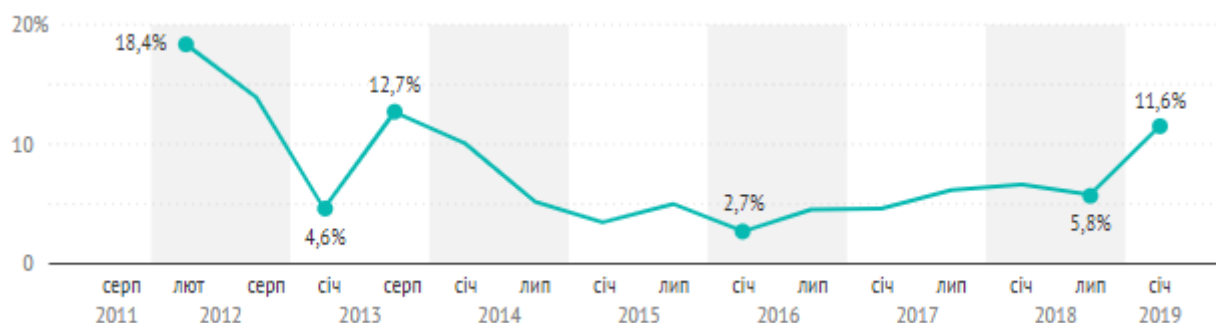


Рисунок 2 – Відносні показники темпів зростання

Ми з впевненістю можемо сказати, що компанії будуть набирати більше і більше працівників. Як наслідок, вони будуть в пошуку автоматизації та



спрощенні співбесід та перевірки або підвищення кваліфікації персоналу.  
(рисунок 3)



Рисунок 2.3 – Плани компаній щодо розширення персоналу на найближчі півроку

## 3.2. ДОСЛІДЖЕННЯ АНАЛОГІВ

### 3.2.1. Опис та аналіз аналогів

Назва	Реєстрація	Можливість отримання результату	Групування по темам	Автоматизованість, проходження за певний час	Можливість вибрати рівень
jsehelper	-	-	+	-	-
quizful	+	+	+	+	-
proghub	+	+	-	+	-
intester	+	+	-	+	-

Таблиця 1 – Дослідження аналогів

#### Jsehelper

Jsehelper – це блог, на якому можна знайти набір питань, для технології і відповіді на них. Але не має реєстрації, та можливості вибирати рівень особи та генерацію питань. Також, великий мінус є те, що не має можливості вести статистику відповідей.

#### Quizful

Quizful – це повноцінний онлайн сервіс для проходження різних тестів. Можна вибрати тести, тему, та дивитись статистику відповідей. Мінусом

даного сервісу є неможливість генерації рандомної кількості питань, ви можете створити тести, але в них буде фіксована однакова кількість питань.

### **Proghub**

Proghub – онлайн сервіс який тільки з'явився. Можна зареєструватись, і проходити різні тести по наявним технологіям. Проблема полягає в тому, що на даному сервісі, не можна вибрати рівень, або кількість питань. Ви просто вибираєте технологію, і проходите відповідний тест.

### **Intester**

Intester – онлайн сервіс для перевірки знань. Є можливість реєстрації, та проходження тестів по певних категоріях. Ви не можете вибрати рівень кваліфікації, та кількість тем, які будуть в даних тестах.

### **Переваги даного проекту, після аналізу даного матеріалу.**

- 1) Реєстрація.
- 2) Можливість подивитись результат.
- 3) Групування питань по тематиці.
- 4) Вибір рівня кваліфікації для якої буде генеруватись список питань.
- 5) Створенням тестів для студентів будуть займатись відповідні люди, і прості студенти не можуть ніяк впливати на результат, або генерувати тести для себе.

### **Недоліки.**

- 1) Потрібна реклама.
- 2) Великий проект не можливо написати без команди.

### **3.2.2. Індивідуальне завдання**

Завданням даної роботи є реалізація web-додатку на тему «система тестування рівня знань студента».

Програма повинна бути простою, інтуїтивно зрозумілою та комфортною у використанні. Програма повинна відповідати системним вимогам.

Додаток має містити базу даних для зберігання користувачів, питань, відповідей на питання.

Також, створенням тестів для користувачів повинна займатись людина з правами admin.

Прості користувачі можуть тільки проходити тести та дивитись свій результат.

## ВИСНОВОК

Практика дозволила застосувати набуті в університеті знання та навички в умовах підприємства ІТ-галузі та здобути новий досвід. Під час проходження практики було виконане індивідуальне завдання. Отриманий програмний компонент, буде основою для написання глобальнішої системи.

Досвід роботи із HTML + CSS + JavaScript та навички створення макетів сторінок і їх стилізації дуже знадобилися при вивченні розробці системи. Також було набуто нові навички роботи із фреймворком Angular та мовою програмування TypeScript. В результаті був створений веб-додаток для тестування студентів.

Практика надала можливість попрацювати у сучасній компанії та отримати досвід співпраці із висококваліфікованими фахівцями, що поділилися досвідом та навчили розробляти web-додатки для реального використання на підприємстві за допомогою фреймворку Angular.