

Запропоновано огляд інформаційних джерел в мережі інтернет, що спеціалізуються в створені бізнес-планів для підприємницької діяльності. Проведено огляд та порівнянний аналіз сучасних інформаційних технологій, що використовуються при розробці веб-орієнтованого програмного продукту. Проведено ознайомлення з можливостями хмарної платформи Microsoft Azure та з використанням хмарних інформаційних технологій розроблено прикладний програмний продукт та відповідна база даних для покрокового формування бізнес-плану підприємства.

It prompted a review of information sources on the Internet that specialize in creating business-plans for entrepreneurial activity. The review and comparative analysis of modern information technologies used in the development of web-based software product. Conducted to familiarize yourself with Microsoft Azure cloud platform and using cloud information technologies developed an application software product and an appropriate database for step-by-step formation of the business plan of the enterprise.

Зміст

Вступ.....	5
Розділ 1. Опис існуючих програмних продуктів.	6
1.1. Програмні продукти	6
1.2 Структура бізнес-плану.....	15
Розділ 2.Опис інтернет технологій.....	19
2.1 Інтернет технології	19
2.2 Хмарні технології	21
Розділ 3. Створення прикладного програмного продукту для складання бізнес-плану підприємства	26
3.1 Ознайомлення з хмарною платформою Microsoft Azure.....	26
3.2 Створення Web додатку з використанням бази даних MySQL	34
3.3 Розробка бази даних	38
3.4 Опис Веб-ресурсу.....	43
Висновки	45
Список джерел.....	46
Додатки.....	47

Вступ

Підприємницька організація починається зі складання бізнес-плану тобто системи менеджменту, спрямованого на вдосконалення роботи будь-якої фірми, незалежно від сфери її діяльності. Метою бізнес-плану може бути отримання кредиту, або залучення інвестицій, визначення стратегічних і фактичних орієнтирів фірми. Починаючи свою діяльність, підприємець повинен чітко уявляти потребу на перспективу у фінансових, матеріальних, трудових та інтелектуальних ресурсах, джерела їх отримання, а також вміти чітко розраховувати ефективність використання ресурсів у процесі роботи фірми.

Таким чином стає актуальним створення веб-орієнтованого прикладного програмного продукту автоматизованого складання бізнес-планів. Метою даної роботи є аналіз міжнародних програмних рішень, які спрощують процес бізнес-планування.

В першому розділі проведено огляд та порівняльний аналіз прикладних програмних продуктів, що використовуються при створенні та плануванні діяльності підприємств. На сьогоднішній день є велика кількість як програм, що треба встановлювати, так і онлайн продуктів, що спрощують процес планування.

В другому розділі запропоновано огляд сучасних інформаційних технологій, що використовуються при створенні веб-орієнтованих програмних продуктів та хмарних застосунків.

В третьому розділі описана покрокова реалізація алгоритму створення хмарного сервісу на платформі Microsoft Azure з підтримкою бази даних, що містить необхідних асортиментів товарів та послуг для створення та початку діяльності підприємства.

У висновках представлено досягнуті в роботі результати.

Код виконавчої програми представлено в додатках.

Розділ 1. Опис існуючих програмних продуктів.

1.1. Програмні продукти

На даний момент існує велика кількість як програм та й інтернет ресурсів, які б допомагали починаючому підприємцю відкрити свою власну справу. Недоліками цих ресурсів є їх ціна. Більшість подібних Web-сайтів або комп'ютерних програм у своїх демо-версіях надають лише оглядові матеріали, тим самим заробляючи гроші на розвитку середнього та малого бізнесу в Україні.

Більш того майже всі продукти є закордонними, тобто не орієнтованими на український ринок та систему оподаткування.

Далі представлені деякі міжнародні програми для складання бізнес-плану

1. **Business Plan PL**
2. **COMFAR III Expert**
3. **Project Expert**
4. **Plan Business Intelligent**

Докладний опис програмних продуктів:

1. **Програма Business Plan PL** допоможе користувачеві створити бізнес-план, розробити різні напрямки фінансового прогнозування.

Перевага даної програми полягає в тому, що вона має не складний інтерфейс, але знання економіки та комп'ютера необхідні.

Програма сама здійснює складні розрахунки та опрацьовує внесені дані. Від користувача, вимагається лише введення потрібних цифр. Вона створена спеціально для людей, які не мають великого досвіду в роботі з подібними програмами. З цією програмою можна вільно працювати на комп'ютері з встановленою системою Windows.

Програму можна поділити на дві частини:

- фінансову (розрахункову)
- текстову (пояснювальну)

Пояснювальна частина в свою чергу ділитися теж на дві частини: "План-зразок" та "Макет-керівництво". Перша частина надає можливість вибрати один з готових бізнес-планів, скориставшись другим, користувач сам поетапно складає свій бізнес-план.

У цьому є свої плюси – можна самостійно вибрати відповідний варіант. Так само визначити, на скільки докладно необхідно описувати ті чи інші пункти. Програма докладно підказує, в якій послідовності та як потрібно складати бізнес-план, наводить приклади та варіанти, а користувач сам вже вирішує, що саме має бути занесено в документ. Але якщо не вистачає часу можна просто ввести дані у відповідний готовий приклад і отримає бізнес-план з усіма розрахунками.

2. Програма COMFAR

Даний програмний продукт створений спеціалістами компанії UNIDO, тому він повністю відповідає даній методиці, яку використовують в процесі бізнес-планування такі великі закордонні інвестиційні інститути, як ЄБРР, Світовий Банк і більшість інших компаній. Перевагою програми COMFAR є повна відсутність можливості розбіжностей з потенційним іноземним інвестором, заснованих на незвичній для нього формі представлення результатів бізнес-планування..

Зручність роботи з COMFAR полягає в тому, що оболонка ретельно продумана та досить зручна для користувача. Зручно відображається результуюча інформація у вигляді таблиці, діаграм та пояснення до них.

Блок опису податків в програмі відсутній, що є основним недоліком програми COMFAR. Не зовсім зрозуміло чому даний блок відсутній, мабуть, ряд

проведених розробниками спрощень спрямовані на універсалізацію програми, що пішло на шкоду функціональності.

Досить багато незручностей доставляє користувачам сервісна підтримка продукту КОМФАР.

3. Програма Project Expert

Програму Project Expert можна назвати самою функціональною та зручною. Для професіоналів в складанні бізнес-планів.

Вся робота системи виконана в простому та зрозумілому вигляді навіть для непрофесіонального користувача, що має загальне розуміння побудови бізнес-плану. Вихідні дані задаються поетапно, тому мінімізується ризик пропуску введення даних.

Функція "Стартовий баланс" надає користувачеві широкі можливості, дозволяючи розробляти інвестиційні проекти будь-якого рівня складності, оперуючи початковими показниками діючого підприємства.

Перевагою Project Expert над системою COMFAR є гнучко спроектований блок податків. Його універсальність дозволяє легко задати як базу оподаткування, так і всі необхідні податки.

Ще однією зручною особливістю Project Expert є добре продумана система роботи з податком на додану вартість. Передбачені в програмі різні схеми обліку і списання ПДВ забезпечують коректність розрахунків навіть в складних і заплутаних випадках.

Складання мережевого плану – ще одна можливість Project Expert. Дана функція системи дозволяє успішно проводити розрахунок інвестиційних проектів, що допускають комбінацію пов'язаних і паралельних етапів інвестування. Незважаючи на громіздкість інтерфейсу цього процесу, результат того вартий.

Даний функціонал є далеко не у всіх програм для складання бізнес-планів і в Project Expert він організований добре.

Труднощі виникають зі сприйняттям внутрішньої логіки блоку "Операційний план". Взагалі, маючи на увазі складність самого програмного продукту і розв'язуваних їм завдань, не можна очікувати від програми простоти та зрозумілості в усьому.

Слабким місцем Project Expert багато фахівців називають розділ пов'язаний з фінансуванням. Організація використання позикових коштів тут представлена у вигляді отримання деякого числа кредитів за фіксованою ставкою на заданий термін, що значно ускладнює роботу із засобами, залученими за позиками типу відкритих кредитних ліній..

Процес формування звітів дозволяє видрукувати таблиці та графіки або експортувати їх у Word. Вихідні форми надають можливість налаштування шрифтів, але досить великий їх розмір у форматі Word народжує необхідність редагувати їх додатково в зовнішньому текстовому редакторі.

Project Expert є закритою системою - користувач не може змінювати розрахунковий алгоритм самостійно.

4. Plan Business Intelligent

Зрозумілий та простий функціонал дозволяє користувачеві швидко розібратись у всіх функціях системи та почати роботу. Для використання програми не потрібні особливі економічні знання тому, що вона є інтуїтивно зрозумілою.

В програмі вміщує описи та практичні поради, також є готові шаблони документів та тези, завдяки чому спрощується робота з програмою.

При створенні бізнес-плану пропонуються макети та описи текстової частини документа, фінансові розрахунки, можливість створювання презентації проекту. Усі розділи знаходяться в окремих документах, фінансові блоки – наведені в таблицях та підготовлені до друку. Програма побудована так, що необхідно ознайомитись з описами розділів та ввести лише свої вхідні данні в шаблони розділу плану.

Ключові переваги програми Plan Business Intelligent:

- Покроковий планувальник.
- Докладні інструкції та описи кожного розділу.
- Макет, шаблони документів.
- Всебічне планування бізнесу, що включає маркетинг, продажі, виробництво, організаційну структуру, розрахунок інвестиційного бюджету і прогноз фінансових результатів.
- Проведення комплексного фінансового аналізу.
- Автоматизовані електронні таблиці, графіки та діаграми.
- Складання презентації бізнес-плану.
- Простий і зручний інтерфейс програми.
- Робота в загальнодоступних форматах MS Word і MS Excel
- Підготовка до друку готових документів.
- Довідкові матеріали про інвесторів і посилання на компанії, де користувач може отримати фінансування.
- Мінімальні системні вимоги до комп'ютера.
- Конкурентна ціна.

Використовуючи програму для складання бізнес-плану Plan Business Intelligent користувач отримує:

- Професійний Бізнес-План.

- Якісну презентацію проекту.
- Комплексний фінансовий прогноз.

Сьогодні інтернет-сервіси набувають великої популярності, бо не потребують встановлення на комп'ютері. Тому існують й інтернет-сервіси, що допомагають планувати бізнес-ідеї. Одними з таких сервісів є E-planificator, BPE24 та інші.

Онлайн конструктора E-planificator

E-planificator - це програма, створена з метою максимально спростити процес введення і виведення інформації при бізнес-плануванні та фінансовому моделюванні та описати за допомогою вступних форм переважна більшість економічних процесів, що виникають при функціонуванні реального сучасного підприємства.

E-Planificator виконаний у формі сайту, в закритій частині користувач може створювати бізнес-плани.

Програма працює за принципом конструктора: користувач вибирає елементи свого бізнес-плану з тих, що йому пропонуються.

Можливості Онлайн конструктора E-planificator по складанню бізнес-планів, фінансових моделей і прогнозів

У особистому кабінеті є можливості:

- розробка проекту колективно, з різних комп'ютерів.
- розробка бізнес-плану повного циклу об'ємом до двохсот-трьохсот листів, основну частину при цьому складуть згенеровані конструктором діаграми та таблиці фінансового розділу бізнес-плану.
- розробляти дванадцять проектів одночасно, переключаючись між ними за потреби.

- вводити вартісні показники в обраній основній валюті, а також в шести додаткових валютах, курси яких по стосовно основної можна задавати та редагувати;
- вибрати (після вивчення розрахованих для вас Конструктором прибутків і грошових потоків) оптимальний для вашого проекту режим оподаткування з п'яти вбудованих: загальну систему оподаткування, спрощену (обкладаються доходи мінус витрати), спрощену (обкладаються доходи), спеціальний режим з єдиним податком на поставлений дохід або патентну систему оподаткування.
- моделювати бюджетні субсидії різного виду і завчасну (до року) закупівлю ресурсів, враховувати інфляцію, аванси, комерційне кредитування, переоцінку і продаж основних засобів та інших необоротних активів, скористатися наявними можливостями автоматичне заповнення.
- зберегти підготовлений бізнес-план у форматі Word; документ у форматі Word завжди може бути збережений в форматі PDF або будь-якому іншому доступному форматі. Таблиці для подальшої роботи додатково можуть бути збережені в форматах Excel або Word.

Онлайн-сервіс ВРЕ24

ВРЕ24 (Business Plan Expert) - це онлайн-сервіс, що дозволяє самостійно виконати необхідні фінансові розрахунки і скласти готовий бізнес-план, навіть не буди фахівцем. Використовується для довгострокового інвестиційного кредитування малого і середнього бізнесу.

Можливості ВРЕ24:

- 10 послідовних кроків, необхідних для створення бізнес-плану і зрозумілих навіть непрофесіоналу;

- Фінансові звіти з прогнозами та показники ефективності в форматі, зрозумілому для банків та інвесторів;
- Структура Бізнес-плану, що включає всі необхідні розділи, приклади та підказки для його написання;
- Оцінка вартості проекту та потреби у власних коштах, кредитах та субсидії
- Аналіз ризиків проекту, його чутливості до зміни ключових параметрів (ціни продуктів, прямі витрати, обсяг збуту та ставка дисконтування)
- Розробка якісних документів, їх зберігання та ін.

PlanBusiness Pro

PlanBusiness Pro - цей покроковий сервіс дозволить легко систематизувати необхідну для бізнес-плану інформацію та скласти професійний документ, який можна представити у банк, інвестору або партнеру по бізнесу.

Щоб почати користуватися сервісом, зовсім не обов'язково бути експертом в стратегічному плануванні.

Послідовно ознайомившись з описами розділів бізнес-плану, користувач заповнює майже закінчені макети документів, адаптуючи внесену інформацію для бізнесу. Таким чином, можна врахувати всі складові та не пропустити нічого важливого.

Залежно від потреб користувача бізнес-план може складатися з описової частини, блоку фінансових розрахунків та презентації проекту. Описова частина також містить резюме документа, яку зазвичай дивляться в першу чергу.

Якщо ж потрібно просто спрогнозувати фінансовий результат нової ідеї бізнесу, не обов'язково складати описову частину і презентацію проекту, якщо його немає кому презентувати. Досить заповнити лише кілька таблиць та

отримати уявлення про потенційний прибуток, точку беззбитковості та інші ключові фінансові показники.

Веб-додаток Enloop

Enloop - веб-додаток для написання бізнес-плану. Сервіс автоматично записує бізнес-план.

Enloop дозволяє передавати фінансову інформацію і бізнес-аналіз. Дозволяє швидше написати бізнес-план, відредагувати і відформатувати відповідно до вимог. Надає автоматизовані прогнози та графіки. Плюс можна ділитися планами і працювати над ними спільно.

Можливості Enloop:

- Фінансові звіти.
- Інтерактивне створення PDF.
- Enloop Performance Score.
- Запрошення.
- Служба підтримки.
- Мультивалютна підтримка.
- Звітний бланк.
- Графіки.

1.2 Структура бізнес-плану

Структура бізнес-плану - письмовий опис проекту, суми та терміни його реалізації. Розділи бізнес-плану багато в чому залежить від того, в якій сфері знаходиться проект, а також, які розміри він має.

Отже, можна розглянути таке поняття, як детальна структура розділів бізнес-плану. Вона включає в себе ряд елементів, які мають власну важливість. (Рис.1.2.1)

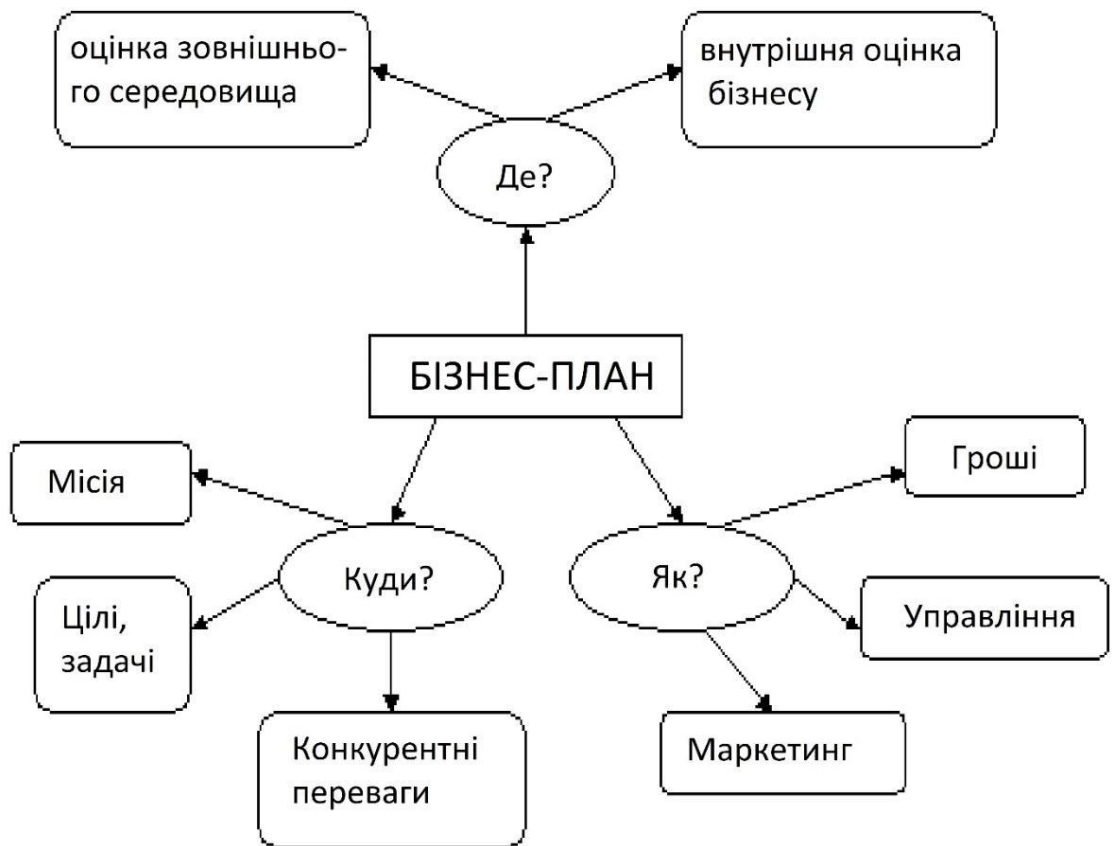


Рис. 1.1.

Структура бізнес-плану може бути представлена такими пунктами:

1. Титульний лист;
2. Резюме;
3. Мета проекту;
4. Аналіз ринку в певній сфері;
5. Положення підприємства в галузі;
6. Опис продукту;
7. Маркетинг план по збуту продукції;
8. План виробництва;
9. Організаційний план;
10. Фінансове планування;
11. Аналіз ризику;

Розкриття найважливіших пунктів:

Положення підприємства в галузі

- усі види діяльності підприємства, його продукція(послуги), споживачі(клієнти);
- організаційно-правова форма підприємства, організаційна структура, засновники, партнери та персонал;
- фінансово-економічні показники діяльності підприємства (фірми);
- розташування підприємства (адреса, характеристика будівлі або приміщення, власність або оренда, необхідність реконструкції);
- специфіка роботи (сезонність, час роботи (години та дні тижня) та інші особливості, пов'язані зі сферою бізнесу або використовуваними ресурсами).

Опис продукту

Орієнтовна структура розділу:

- найменування продукції (послуги);
- призначення та область застосування;
- опис і основні характеристики;
- конкурентоспроможність продукції (послуги) з виділенням параметрів, за якими продукт перевершує конкурентів або поступається;
- патентоспроможність і авторські права;
- наявність або необхідність ліцензування продукції;
- наявність сертифіката якості продукції;

План виробництва

У цьому розділі здійснюється розрахунок виробничих витрат на планований обсяг збуту: розрахунок собівартості продукції, складання кошторису витрат на виробництво, визначення змінних і постійних витрат в собівартості. Орієнтовна структура розділу така:

- загальні відомості про виробничі структури підприємства, наявності необхідних транспортних зв'язків, інженерних мереж (електроенергія, вода, тепло, каналізація, зв'язок та ін.), ресурсів, близькість до ринку збуту;
- використовувана технологія і рівень її освоєння підприємством;
- потреба в виробничих площах;
- потреба в виробничому персоналі (чисельність, спеціалізація, кваліфікація);
- задоволення вимог щодо забезпечення екологічності виробництва для навколишнього середовища і безпеки працюючих;
- готівкові та необхідні виробничі потужності;
- характеристика обладнання;
- планування джерел постачання сировини, матеріалів, обладнання та робітничих кадрів;
- визначення собівартості усіх видів продукції;

- кошторис поточних витрат на виробництво;
- аналіз структури собівартості.

Фінансове планування

В цьому розділі плануються інвестиційні вкладення (на будівельно-монтажні роботи, придбання обладнання, створення оборотних коштів та ін.).

Обираються джерела фінансування проекту. Наводиться схема повернення кредитних коштів.

Завершуватися розділ фінансового планування має аналізом ефективності проекту. Для цих цілей можна використовувати методику проектного аналізу, методи економічного аналізу фінансово-господарської діяльності (рентабельність, фінансова стійкість проекту тощо).

Орієнтовна структура розділу така:

- звіт про прибутки та збитки за кожен рік терміну реалізації проекту, (перший рік помісячно);
- структура податкових платежів;
- план руху грошових коштів першого року;
- плановий баланс першого року;
- потреба в інвестиціях;
- витрати, пов'язані з обслуговуванням кредиту (лізингу) (можуть бути проаналізовані окремо);
- аналіз ефективності проекту.

Розділ 2.Опис інтернет технологій.

2.1 Інтернет технології

Мова розмітки гіпертексту (HTML) - це стандартна мова розмітки для створення веб-сторінок та веб-програм.

Веб-браузери отримують HTML-документи з веб-сервера або з локального сховища та перетворюють документи на мультимедійні веб-сторінки. HTML описує структуру веб-сторінки семантично та включає сигнали для появи документа.

HTML-елементи є блоками HTML-сторінок. За допомогою конструкцій HTML, зображення та інші об'єкти, такі як інтерактивні форми, можуть бути вбудовані у відображену сторінку. Елементи HTML позначені тегами, написаними за допомогою кутових дужок. Браузери не відображають теги HTML, але використовують їх для інтерпретації вмісту сторінки.

MySQL - це система управління реляційною базою даних з відкритим кодом (RDBMS). MySQL був розроблений для підвищення швидкодії обробки великих баз даних. Ця система була створена як альтернатива комерційним системам. MySQL з самого початку була дуже схожою на mSQL, проте з часом вона все розширювалася і зараз MySQL — одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

PHP- це серверна скриптова мова, призначена для розробки веб-сайтів, а також використовується як мова програмування загального призначення.

PHP-код може бути вбудований в HTML-код, або його можна використовувати в поєднанні з різними системами веб-шаблонів, системами

управління веб-контентом та веб-структурами. PHP-код, як правило, обробляється інтерпретатором PHP, реалізованим у вигляді модуля на веб-сервері або як виконуваний файл інтерфейсу Common Gateway Interface (CGI). Веб-сервер поєднує в собі результати інтерпретованого та виконаного PHP-коду, який може бути будь-яким типом даних, включаючи зображення, зі створеної веб-сторінкою. PHP-код також може бути виконаний за допомогою інтерфейсу командного рядка (CLI) і може бути використаний для реалізації автономних графічних програм.

Стандартний інтерпретатор PHP, який підтримує Zend Engine, є вільним програмним забезпеченням, випущеним під ліцензією PHP.

2.2 Хмарні технології

Хмарні технології — це технології, які надають користувачам Інтернету доступ до ресурсів сервера і використання програмного забезпечення як онлайн-сервісу.

Хмарні технології поширені серед малих підприємств, установ, які не мають змоги придбати і обслуговувати власні сховища даних або опрацювання складних обрахунків.

Хмарні системи надають такі види послуг:

Storage-as-a-Service ("зберігання як сервіс")

Storage-as-a-Service уявляє з себе дисковий простір, що можна використовувати при потребі.

Storage-as-a-Service дає можливість зберігати дані в зовнішньому сховищі, в "хмарі". Для користувача, це виглядає, як додатковий логічний диск або папка. Прикладом може служити Google Drive та інші схожі сервіси.

Database-as-a-Service ("база даних як сервіс")

Даний вид сервісу більш цікавий для розробників, він дозволяє працювати з базою даних не встановлюючи додаткового програмного забезпечення. Також в цьому випадку легше надається доступ до проекту різним виконавцям.

Information-as-a-Service ("інформація як сервіс")

Сервіс дозволяє віддалено користуватись різними видами інформації, що може змінюватись кожної хвилини.

Process-as-a-Service ("управління процесом як сервіс")

Сервіс являє собою ресурс, що може використовуватись для об'єднання декількох ресурсів (послуги або дані, що знаходяться в одній "хмарі", або в інших доступних "хмар"), для створення одного бізнес-процесу.

Application-as-a-Service ("додаток як сервіс")

Також має назву Software-as-a-Service ("ПО як сервіс"). Позиціонується як «програмне забезпечення за потреби» яке розташоване на віддалених серверах та будь-який користувач може мати до нього доступ за допомогою Інтернету, а оновленням та ліцензією на дане забезпечення регулює постачальник послуги.

Як приклад можна привести Google Docs, Google Calendar та інші онлайн-програми.

Platform-as-a-Service ("платформа як сервіс")

У використанні користувачеві надається комп'ютерна платформа з встановленою операційною системою та деяким програмним забезпеченням.

Integration-as-a-Service ("інтеграція як сервіс")

При використанні даної послуги користувач отримує повний інтеграційний пакет, програмні інтерфейси між додатками та управління їх алгоритми. Сюди входить послуги і функції пакетів централізації, оптимізації та інтеграції корпоративних додатків (EAI).

Security-as-a-Service ("безпека як сервіс")

Дана послуга надає можливість користувачам швидко розгортати продукти, що забезпечує безпечне використання веб-технологій, електронного листування, локальної мережі. Це дозволяє користувачам економити на розгортанні та підтримці своєї власної системи безпеки.

Management / Governace-as-a-Service ("адміністрування та управління як сервіс")

Користувач має змогу управляти та задавати характеристики одного або багатьох "хмарних" сервісів.

Infrastructure-as-a-Service ("інфраструктура як сервіс")

Сервіс надає в користування комп'ютерну інфраструктуру. У більшості віртуальні комп'ютери (платформи), що об'єднані в мережу, а користувач має змогу налаштовувати їх під власні цілі.

Testing-as-a-Service ("тестування як сервіс")

Дає можливість тестування локальних або "хмарних" систем з використанням тестового програмного забезпечення з "хмари".

Схема класифікації сервісів за типом послуг(Рис 2.1):



Рис.2.1

Хмарні технології за формою власності бувають:

- Публічні
- Приватні
- Гібридні.

Публічна хмара - це ІТ-інфраструктура, що надає одночасний доступ безліч компаній і сервісів. Обслуговування та управління здійснює керівництво ресурсу.

Прикладами можуть служити онлайн-сервіси: Amazon EC2, Google Apps / Docs, Microsoft Office Web.

Приватна хмара - це захищена ІТ-інфраструктура, що використана в інтересах однієї організації. Інфраструктура може розміщуватися або в приміщеннях замовника, або у зовнішнього оператора (або частково у замовника і частково в оператора).

Гібридна хмара - це ІТ-інфраструктура об'єднує та використовує кращі якості двох попередніх типів хмар. Даний тип хмари найчастіше використовується при сезонній активності підприємства.

Переваги хмарних обчислень:

- Доступ та опрацювання інформації з будь-якого пристрою підключеного до Інтернету;
- Одночасний доступ до інформації декільком користувачам;
- Збереження даних при поломці пристрою(ПК, планшетом, телефоном)
- Використовується самі останні версії програм;

Недоліки:

- Постійне з'єднання з Інтернет;
- Обмеження в розгортанні програмного забезпечення
- Не захищеність особистий даних користувача, що зберігаються в “хмарі”

Розділ 3. Створення прикладного програмного продукту для складання бізнес-плану підприємства

3.1 Ознайомлення з хмарною платформою Microsoft Azure

Microsoft Azure – хмарна платформа Microsoft. Надає широкий спектр послуг такі, як розробка та виконання програм, зберігання даних на віддалених серверах. Microsoft Azure пропонує дві моделі - платформи як сервісу (**Platform-as-a-Service**) та інфраструктури як сервіс (**Infrastructure-as-a-Service**).

Мережа глобальних дата-центрів Microsoft забезпечують працездатність платформи Microsoft Azure.

Особливості даної моделі:

- оплата тільки спожитих ресурсів;
- загальна, багатопотокова структура обчислень;
- абстракція від інфраструктури.

Для кожного проекту здійснюється запуск віртуальної машини, це є основою роботи платформи Microsoft Azure. Користувач самостійно визначає обчислювані потужності та необхідний обсяг даних. При потребі платформа може виділити додаткові ресурси дата-центру.

Microsoft Azure як PaaS забезпечує:

- виділення ресурсів на вимогу;
- обробку помилок інфраструктури для постійної доступності;

Microsoft Azure як інфраструктура реалізує можливість оренди таких ресурсів, як сервери, пристрої зберігання даних та мережеве обладнання. Постачальник послуги управляє всією інфраструктурою, користувач управляє тільки операційною системою і встановленими додатками. У галереї образів доступні образи наступних операційних систем: Windows Server

(2008, 2012 Technical Preview), CoreOS, Ubuntu Server, CentOS, openSUSE, SUSE Linux Enterprise Server, Oracle Linux.

Microsoft Azure складається з:

- Compute - компонент, який забезпечує обчислення на платформі Windows Azure.
- Storage - компонент сховища. Сховище не має можливості використовувати реляційну модель і є альтернативною, «хмарної» версією SQL Server.
- Fabric - Azure Fabric є ядром та «контролером» платформи, виконує функції моніторингу, забезпечення відмовостійкості, виділення потужностей, розгортання серверів, віртуальних машин і додатків, балансування навантаження та управління обладнанням.

Windows Azure - потужна «хмара», служба для розробки та обчислень. Користувач самостійно формує віртуальні машини, як налаштовані, так і не настроєні, з внутрішнім компонентом SQL Server. Забезпечена доступність і надійність, зручний інтерфейс і інструменти командного рядка для всіх операційних систем. Розгортання віртуальних мереж і сховищ даних є простою процедурою. Легко розгорнути і інші служби, такі як Active Directory, мобільні служби, шина обслуговування, SQL Server підготовка звітів, веб-сайти тощо, для отримання потужної IaaS.

Для початку роботи з хмарною платформою Microsoft Azure потрібно пройти реєстрацію.

Azure надає можливість створити студентську підписку використовуючи адресу університетську електронної пошти, але на жаль вона має обмежений набір безкоштовних функцій, також є можливість оформити Free Trial підписку. (Рис. 3.1)

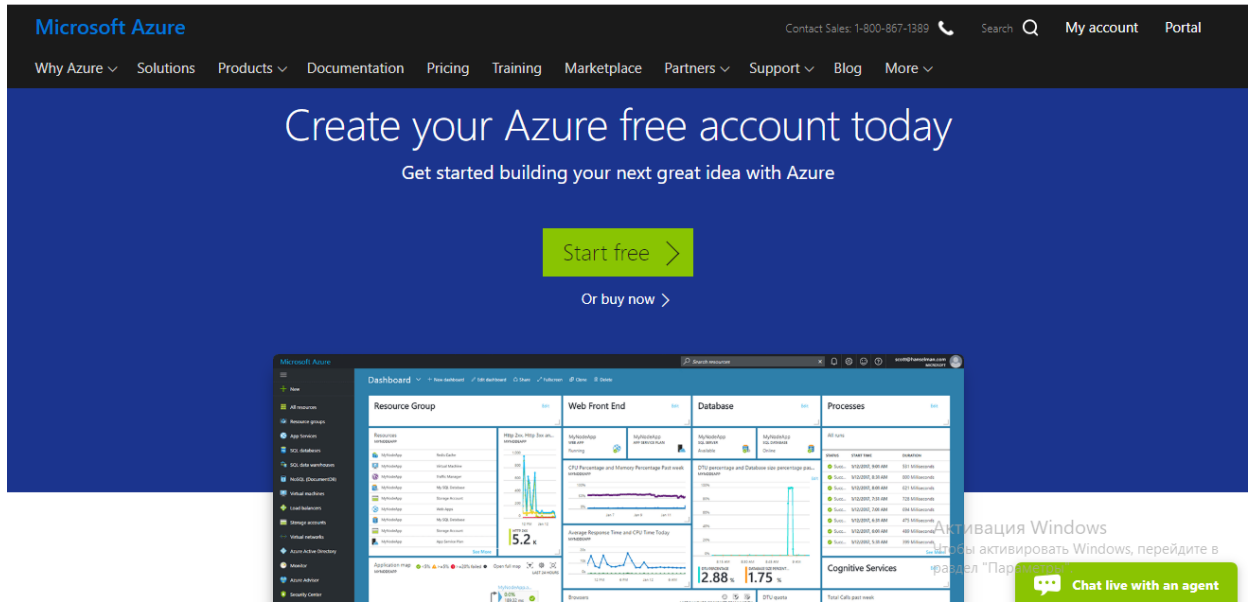


Рис. 3.1

В якій ви отримуєте можливість використання ресурсів платформи, бонусну суму на ваш рахунок, яку можна використовувати протягом 30 днів, також протягом 12 місяців вільно користуватись безкоштовними функціями хмари. (Рис. 3.2)

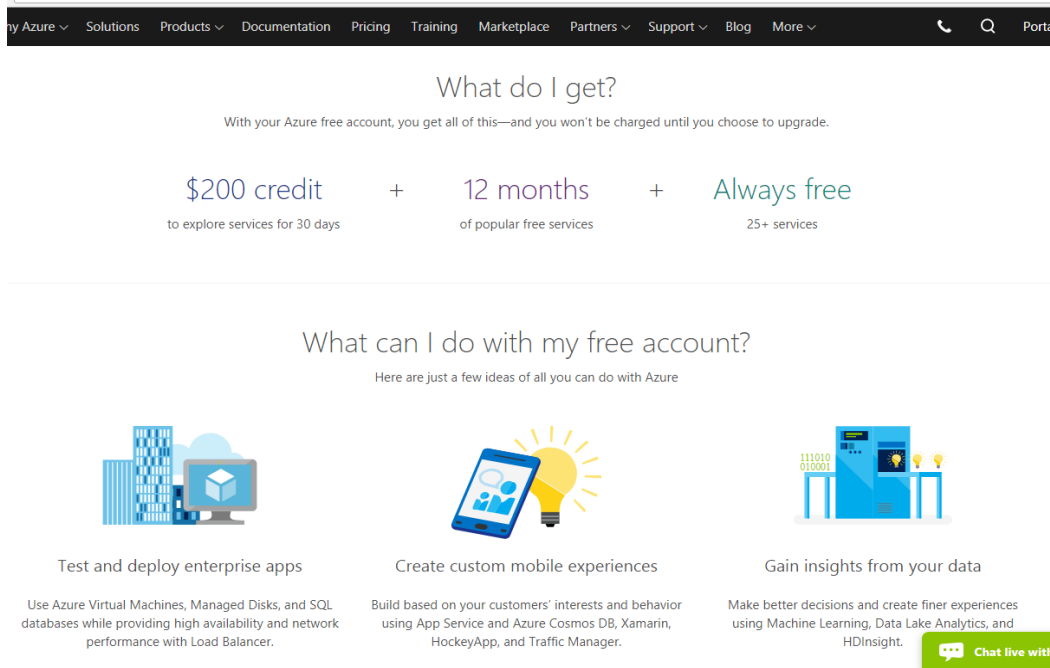


Рис. 3.2

Для реєстрації потрібно мати, або створити новий обліковий запис Microsoft. На наступному кроці потрібно ввести свої персональні дані та данні своєї кредитної карти для того щоб після закінчення безкоштовної підписки можна було продовжити платну підписку. (Рис. 3.3)

Account verification

Enter your credit card number so we can verify your account. You won't be charged for the subscription.

Choose a payment method

New payment method

☒ Credit card

Add payment information

Credit card type *

☒ VISA ☐ MasterCard ☐ American Express ☐ Discover

Credit card number *

4111111111111111

Expiration date *

01 2018

Name on card *

Bret Tester

CVV *

123 What's this?

Рис. 3.3

Хмарна платформа Azure надає великий функціонал для розробників.

(Рис. 3.4)

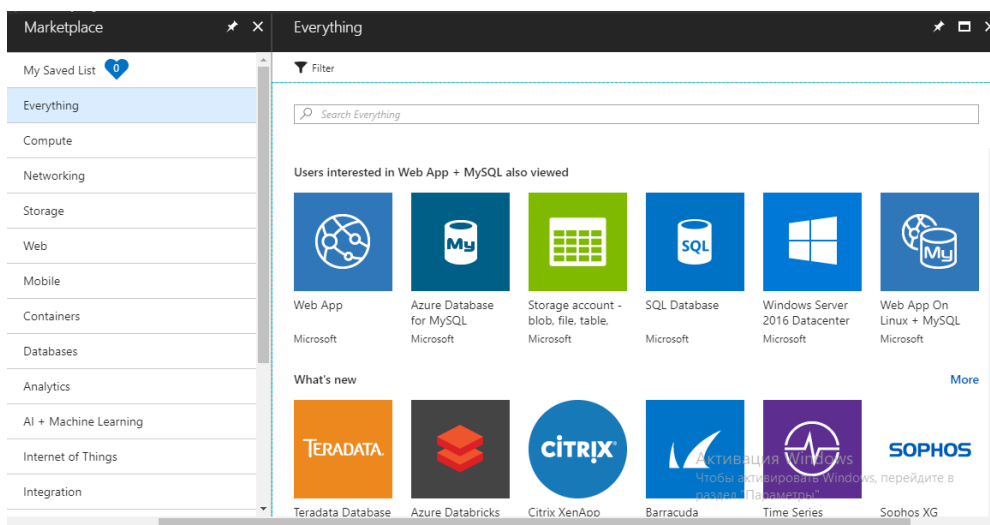


Рис. 3.4

Існують три сфери сервісів Microsoft Azure — сфера **обчислень**, на основі сервісів якої працюють майже всі інші сервіси, сфера **інтеграції** — для мобільних додатків, сфера **роботи з даними** — всі сервіси, які дають дозвіл різними можливостями взаємодіяти та змінювати дані.

1. Сфера обчислень

Virtual Machines — віртуальна машина. В даній моделі можливий запуск практично будь-яких додатків, встановлені образи операційних систем, інтегрованої в платформу Microsoft Azure. Відмінність віртуальних машин в хмарі від локальних — це деякі обмеження (на кількість відкритих портів, тощо), що впливають на можливість розміщення навантажень, але зроблені вони для мінімізації можливості виходу з ладу сервісу.

Cloud Services оперує рольовою моделлю — рішення поділяються на Web і Worker.

Web-роль — це виділення веб-сервера IIS для розміщення інтерфейсу проекту, яким будуть користуватися користувачі.

Worker-роль — аналог Windows-сервісів. Тут виконується обробка та можлива бізнес-логіка проекту.

Web Sites — можливість розробки веб-сайтів на .NET, Node.js або PHP і Python (або розгортаючи CMS — WebMatrix, Joomla, Drupal, WordPress, DotNetNuke, Umbraco та ін.). Для управління Web-сайтами можна використовувати командний рядок Windows, Mac, Linux.

2. Сфера доступу до даних

Сфера доступу до даних вміщує в себе нереляційні сховища даних: таблиці, диски, черги, зберігання двійкових об'єктів, реляційне сховище даних у вигляді SQL Database.

Таблиці — сховище таблиць використовується додатками. В таблиці зберігаються структуровані дані без відносин між собою.

Черги — забезпечують обмін повідомленнями між додатками.

Блоби — зберігають великі обсяги неструктурованих текстових або бінарних даних.

SQL Database — це масштабована хмарна служба бази даних, побудована на основі технологій SQL Server.

Крім SQL Database, на платформі є ще кілька пропозицій - MySQL і MongoDB, аналогічно доступні як сервіси.

Caching — розподілений кеш пам'яті, з допомогою якого програми отримують швидкий доступ до даних, що зберігаються в оперативній пам'яті;

3. Сфера інтеграції

Media Services охоплює хмарні версії багатьох існуючих технологій платформи мультимедіа Microsoft, в тому числі для перегляду, кодування,

зміна формату і захисту контенту, а також потокової передачі за запитом в реальному часі.

Mobile Services пропонує хмарну інфраструктуру для всіх найвідоміших мобільних платформ: Windows 8+, Windows Phone 8+, iOS, Android, HTML 5. За допомогою наданих користувачу функцій можна реалізувати хмарних бекенд, на який перенести завдання по зберіганню даних, реєстрацію і Push-повідомлень. Підтримується Xamarin.

Identity — служба аутентифікації дає змогу управляти посвідченнями та дає доступ до додатків за допомогою служби Microsoft Azure Active Directory (колишній Access Control Service) можна забезпечити єдиний вхід, підвищену безпеку і просту взаємодію з вже розгорнутими в Active Directory додатками.

Azure Active Directory дозволяє вирішувати завдання єдиної авторизації користувачів для багатьох сервісів (Single Sign On), вести єдиний каталог користувачів, синхронізувати дані каталогу Active Directory тощо.

Service Bus є можливість безпечного обміну повідомленнями та дозволяє створювати розподілені та слабозв'язаних програми в хмарі, а також гібридні програми, розміщені одночасно в приватних і загальнодоступних хмарних службах. Оперує термінами Relay, Topics, Queues, Notification Hubs і Event Hubs.

BizTalk Services — сервіс, який призначений для вирішення завдань інтеграції різнорідних середовищ на рівні підприємства і хмари, пропонуючи можливості Business-to-Business (B2B) і Enterprise Application Integration (EAI) взаємодій.

Traffic Manager — диспетчер трафіку забезпечує балансування навантаження по вхідному трафіку між кількома розміщеними службами

Windows Azure незалежно від того, працюють вони в одному центрі обробки даних або розподілені по декількох.

RemoteApp дозволяє розміщувати в хмарі Azure існують клієнтські Windows-додатки та отримувати до них доступ з будь-яких комп'ютерів, планшетів, ноутбуків, телефонів або через RDP-клієнт (Windows, Mac OS X, iOS і Android).

API Management пропонує розробникам власних API можливість отримати оточення для управління, моніторингу та адміністрування свого API, що розміщений в будь-якому місці, як в хмарі, так і на будь-якому хостингу, включаючи власну інфраструктуру.

3.2 Створення Web додатку з використанням бази даних MySQL

Для створення Web додатку з використанням бази даних MySQL необхідно створити ресурс який буде використовуватися. (Рис. 3.5)

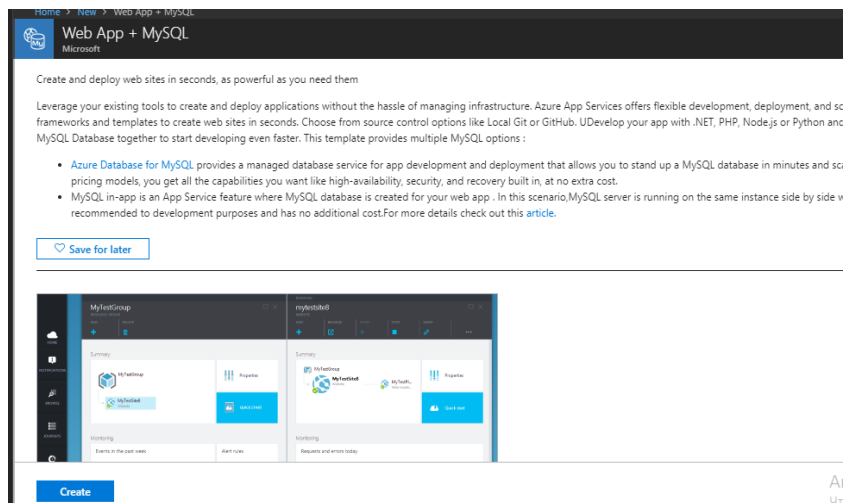


Рис.3.5

Ввести назву додатку, обрати підписку та налаштувати параметри сервера MySQL. В меню якого треба зазначити назву сервера, створити доступ рівня «Адміністратор», обрати версію та назву бази даних. (Рис.3.6)

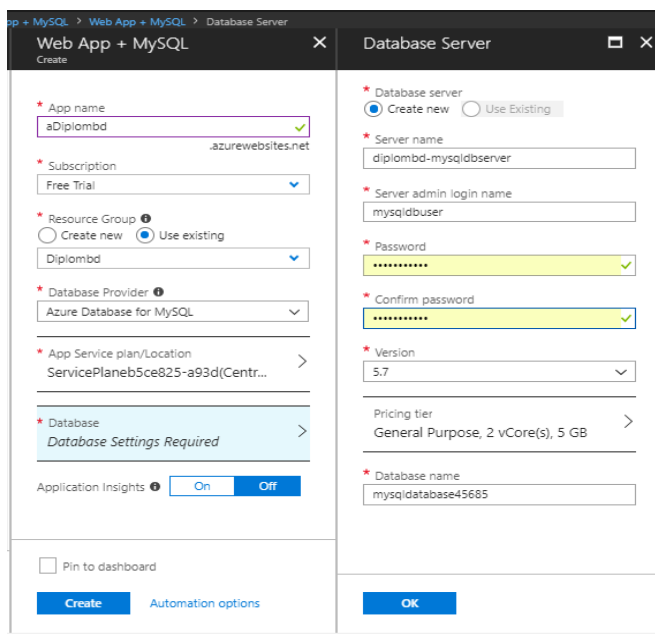


Рис.3.6

Система запропонувала обрати цінову категорію, потужність віртуального сервера, розмір сховища. Також є можливість при обраній ціновій категорії додавати характеристики сервера. (Рис. 3.7)

The screenshot shows the 'Database Server' configuration page in the Azure Portal. The 'Pricing tier' section is active, showing three options: Basic, General Purpose (selected), and Memory Optimized. The 'General Purpose' tier is selected, showing 'Up to 32 vCores with predictable IO performance (2-32 vCores)'. The 'Compute Generation' is set to 'Gen 4'. The 'vCore' slider is set to '2 vCores'. The 'Storage' slider is set to '5 GB'. The 'Price Summary' on the right shows a total monthly cost of 157.08 USD. The 'Database name' is 'mysqldbserver'.

Рис.3.7

Хмарна платформа пропонує різні можливості розміщення файлів проекту.

- Visual Studio Team Services
- OneDrive
- Локальний репозиторій Git
- GitHub
- Bitbucket
- Dropbox
- Зовнішній репозиторій

У свою ж чергу мною було обрано один з найбільших веб-сервісів для спільної розробки програмного забезпечення GitHub. На даному веб-сервісі також

треба пройти процес реєстрації. Після створення особистого кабінету, створюємо свій репозиторій де будуть зберігатись файли проекту. (Рис. 3.8)

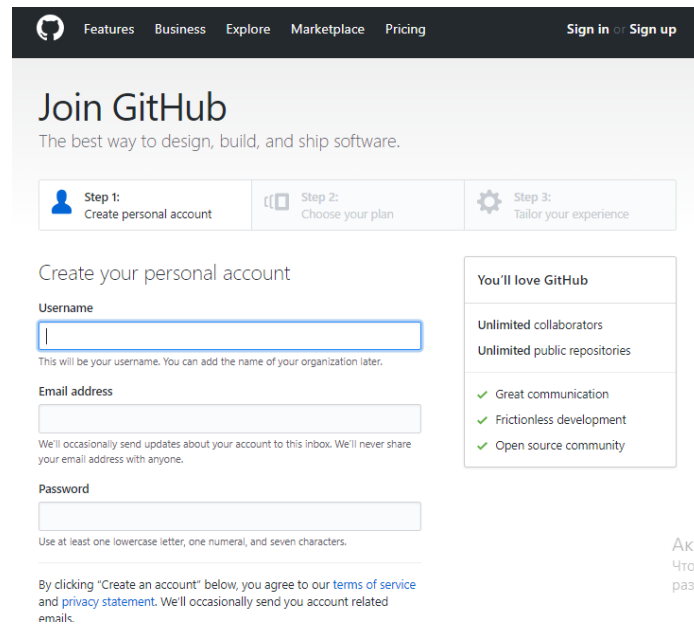


Рис. 3.8

Завантаживши файли проекту в створену папку в хмарі, можна приступити до розгортання додатку. (Рис. 3.9)

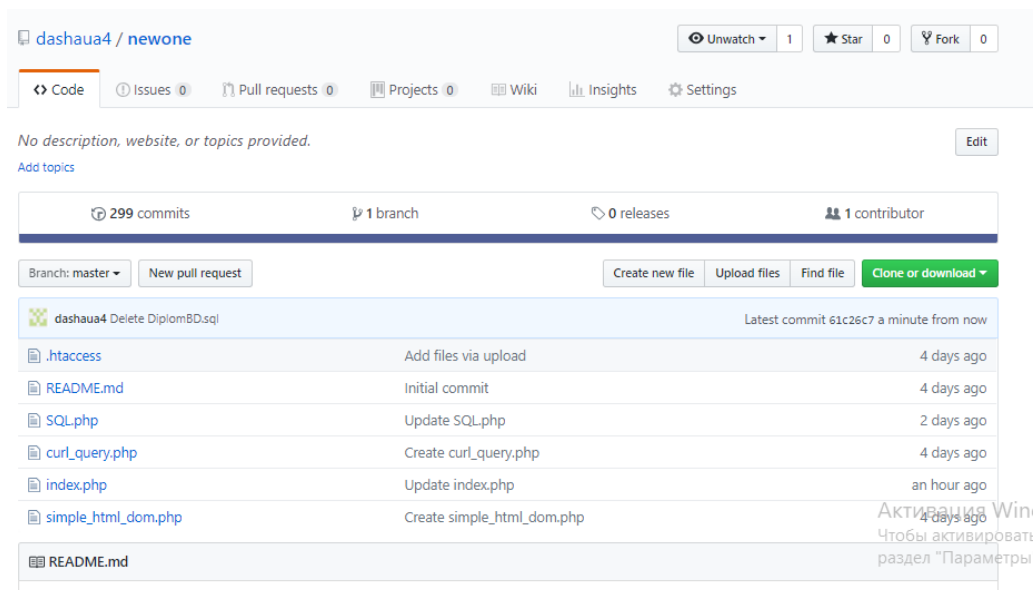


Рис. 3.9

Розгортання додатку відбувається таким чином (Рис. 3.10, 3.11):

1. Вибрати зручний спосіб розгортання проекту;
2. Підтвердити особисті данні;

При обиранні методу розгортання проекту, у даному випадку веб-сервіс GitHub (Рис.3.10).

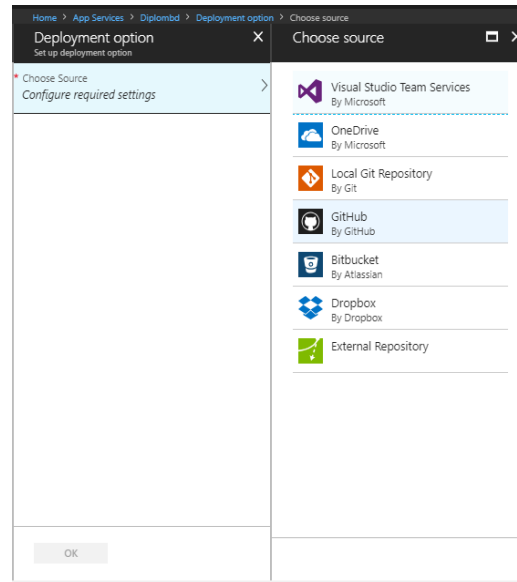


Рис. 3.10

Після обраного репозиторія треба пройти авторизацію та обрати назву проекту в представленому списку проектів, що розміщені на веб-сервісі.

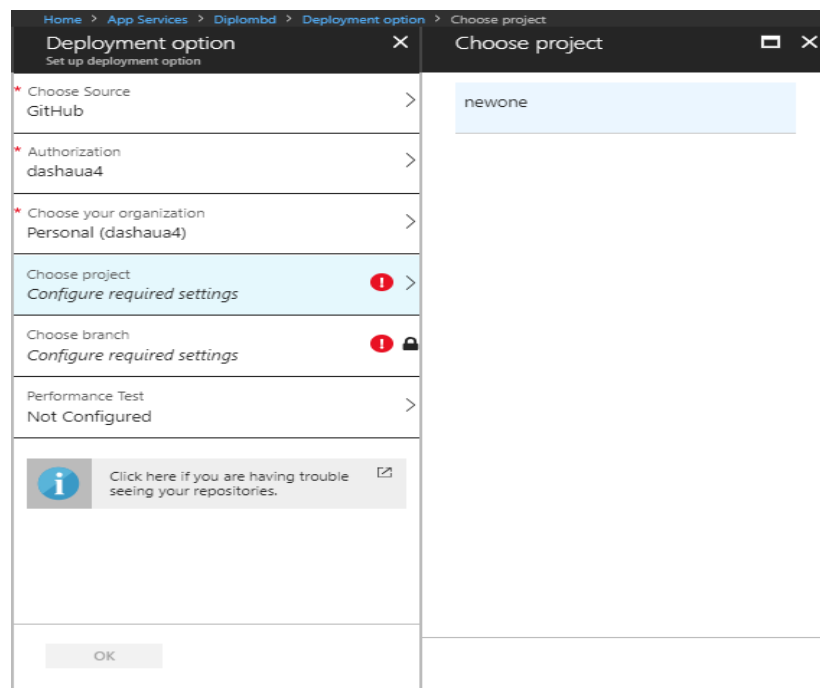


Рис. 3.11

3.3 Розробка бази даних

Проаналізувавши процеси створення підприємства, було зроблено висновки що база даних повинна містити такі розділи (Рис.3.12):

- Обладнання (Equipment):
 - Робочі та графічні системи (WG_system);
 - Офісні комп'ютери (Office_comp);
 - Монітори (Monitors);
 - Периферія (Periphery);
- Офісні меблі (Office furniture)
 - Столи (Tables);
 - Стільці (Chair);
 - Шафи (Locker);
- Персонал (Employees);
- Приміщення (Work_place);
- Поточні витрати підприємства (Податки (Taxes), Комунальні платежі (Public_services), Реклама (Advertisement));
- Зовнішні інвестиції (Creaditing);

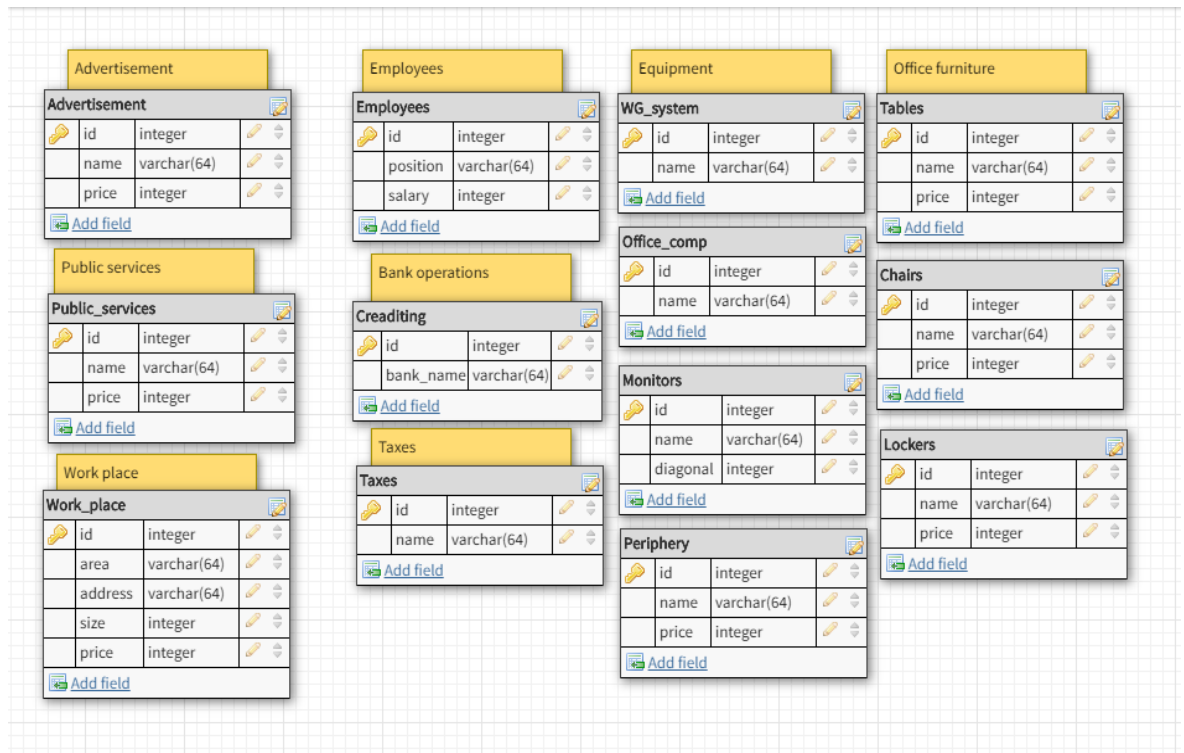


Рис.3.12

Створення таблиць відбувається за допомогою вбудованої функції `Cr_table()`, яка проводить перевірку та підключення до бази даних. Сама база даних знаходиться в хмарному сховищі Microsoft Azure, а опрацювання запитів до неї на створення таблиці у випадку успішного виконання операції підтверджується повідомленням про успішне створення, а в іншому випадку повідомленням про помилку і текст помилки.

```
$sql = "CREATE TABLE Chairs (
id INTEGER AUTO_INCREMENT PRIMARY KEY,
name VARCHAR(64) ,
price INTEGER)";
if ($conn->query($sql) === TRUE) {
    echo "Table Table created successfully";
} else {
    echo "Error creating table: " . $conn->error;
}
mysqli_close($conn);
```

Заповнення таблиць здійснюється при використанні алгоритму простого парсера. Парсер -програма або функція який застосовується для збору та аналізу інформації з веб-ресурсів. Для отримання даних з сторінки у форматі HTML використовуємо функцію `curl_get($url,$brows)`.

Функція ідентифікує користувача у мережі як пошукову систему, а не комерційну IP адресу. Це дає користувачу можливість без перешкод отримувати інформацію з будь-яких веб ресурсів не хвилюючись через блокування. Функція приймає такі параметри, як адресу джерела та назву тої пошукової системи, якою ми хочемо представитись.

```
<?php
function curl_get($url,$referer='http://www.google.com'){
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_HEADER, 0);
curl_setopt($ch, CURLOPT_USERAGENT, 'Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.2
(KHTML, like Gecko) Chrome/22.0.1216.0 Safari/537.2');
curl_setopt($ch, CURLOPT_REFERER, $referer);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$data = curl_exec($ch);
curl_close($ch);
return($data);}?>
```

За допомогою бібліотеки `simpleHTMLDOM` і її функцій `str_get_html()` та `find()`, отримуємо перетворений HTML код згаданої сторінки. За допомогою функції `find()` формуємо масив з потрібними нами значеннями.

На першому етапі формування з допомогою функції `curl_get()` отримуємо сторінку та використовуючи функцію бібліотеки `str_get_html()` завантажуюмо HTML із URL. Розпочинаємо пошук елементів за іменем класу `‘.name_product’`. На наступному кроці читаємо текст елемента та інформацію з характеристик елемента та записуємо потрібні характеристики у масив.

```
include('curl_query.php');
include('simple_html_dom.php');

$html=curl_get('https://meblihit.com.ua/catalog/modul%60na_systema_
ofys/');
$dom=str_get_html($html);
$tables=$dom->find('.name_product');
$i=1;
foreach($tables as $table)
{
    $tobd=array();
    $tobd['id']=$i++;
    $a=$table->find('a',0);
    $tobd['name']="". $a->plaintext. "";
    $one=curl_get('https://meblihit.com.ua'. $a->href);
    $one_dom=str_get_html($one);
    $cost=$one_dom->find('.item_current_price',0);
    $tobd['price']=(int)$cost->plaintext;
    Insert('Tables',$tobd);}
```

Останнім кроком є передача масиву з інформацією в функцію INSERT() яка реалізує запит який заповнює таблицю даними.

```
function Insert($table,$object)
{
    // Create connection
    $conn = mysqli_connect($servername, $username, $password, $dbname);
    // Check connection
    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }
    $columns=array();
    foreach($object as $key=>$value)
    {
        $columns[]=$key;
        $masks[]=$value;
        if($value==null)
        {$object[$key]='NULL';}
    }
    $columns_s=implode(',',$columns);
    $masks_s=implode(',',$masks);
    foreach($masks_s as $value1)
    {echo $value1;
    }
    $sql="INSERT INTO $table ($columns_s) VALUE ($masks_s)";
    if ($conn->multi_query($sql) === TRUE) {
        echo "New records created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }
    mysqli_close($conn);
}
```


3.4 Опис Веб-ресурсу

Створений веб-ресурс дає змогу користувачу дізнатись як спланувати свій бізнес та як самостійно розпочати підприємницьку діяльність. Також існує можливість створити скорочений варіант бізнес-плану прорахувати масштаби вкладень при відкритті своєї справи, рентабельність, період окупності.

Веб-сайт поділяється на 4 сторінки:

- Головна;
- Бізнес-планування;
- Функції;
- Контакти;

Головна

На цій сторінці розташована вступна частина. Коротка інформація про бізнес-планування та можливості сайту.

Бізнес-планування

Детальний опис процесу створення бізнес-плану, етапи тощо. Сторінка призначена допомогти користувачу поглибити знання в галузі планування діяльності та під час самостійного створення бізнес-плану.

Функції

На сторінці представлений калькулятор який дає змогу користувачу створити бізнес-план. На сторінці є можливість обрати три категорії: Дизайн та Реклама, ІТ Компанія, Офіс.

Під кожную категорію програма генерує дані з врахуванням виду діяльності.

Програма надає такі поля вхідних даних:

При створенні необхідно розпочати з визначенням місця де буде існувати підприємство.

- Приміщення

Якщо в користувача є власне приміщення це суттєво зменшує стартовий капітал, якщо користувач немає такої можливості то можна обрати «Оренда», та на цьому кроці також можна обрати відповідні характеристики, наприклад: район та квадратуру приміщення та отримає інформацію про мінімальні та максимальну вкладення в цей розділ.

- Комунальні послуги

Зважаючи на ти приміщення та його розміри програма вираховує коштовність щомісячних комунальних послуг.

- Персонал

У цьому розділі користувач має можливість ввести кількість та посаду персоналу. З урахуванням податків програма згенерує розмір щомісячних витрат.

- Обладнання (Меблі)

- Технічне обладнання :

1. Комп'ютери

2. Монітори

3. Периферія.

- Реклама

Не одне підприємство на етапі формування та під час старту не може обійтись без реклами, тому на сайті є можливість обрати зручну для вас форму реклами, тривалість тощо, та отримати інформацію про цінову політику.

- Інвестиції

Найголовнішою частиною сервісу є Інвестиції. Після обрахунку витрат користувачу буде надана інформація про можливості залучення коштів.

Висновки

1. Запропоновано огляд інформаційних джерел в мережі інтернет, що спеціалізуються в створенні бізнес-планів для підприємницької діяльності.
2. Проведено огляд та порівняльний аналіз сучасних інформаційних технологій, що використовуються при розробці веб-орієнтованого програмного продукту.
3. Проведено ознайомлення з можливостями хмарної платформи Microsoft Azure та створено веб-додаток з підключенням бази даних MySQL.
4. На основі покрокового аналізу формування підприємства з метою його комплектування розроблена відповідна база даних.
5. Створено алгоритм пошуку і обробки інформації з різних інформаційних джерел та автоматичне формування бази даних.

Список джерел

1. <https://docs.microsoft.com/en-us/azure/>
2. <http://php.net/docs.php>
3. <https://dev.mysql.com/doc/>
4. <https://docs.microsoft.com/en-us/azure/cloud-services-php-create-web-role>
5. Веллинг Люк, Томсон Лора. Разработка web-приложений с помощью PHP и MySQL-2017г.
6. Владимир Сафонов. Основы информационных технологий, Платформа облачных вычислений Microsoft Windows Azure- Учебное пособие-2016г.
7. Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS) 1st Edition by Michael J. Kavis-2014.
8. Jaimie Cloud Paperback – 1 Jan 2012. Ronald Cohn Jesse Russell

Додатки

Додаток 1-данні в таблицях бази даних

Table 'Tables'id: 1 - Name: Модульна система Офіс (ofis)-13991
id: 2 - Name: Модульна система Офіс (ofis), стелаж 4х дверний-1994
id: 3 - Name: Модульна система Офіс (ofis), стелаж 3х дверний-1929
id: 4 - Name: Модульна система Офіс (ofis), стелаж 2х дверний-1697
id: 5 - Name: Модульна система Офіс (ofis), стол 160-3392
id: 6 - Name: Модульна система Офіс (ofis), стол углової-658
id: 7 - Name: Модульна система Офіс (ofis), стол бокової-1008
id: 8 - Name: Модульна система Офіс (ofis), шкаф углової-1941
id: 9 - Name: Модульна система Офіс (ofis), пенал для одягу 50-1323
id: 10 - Name: Модульна система Офіс (ofis), шкаф для одягу 60-1595
id: 11 - Name: Модульна система Офіс (ofis), углове окончання-835
id: 12 - Name: Модульна система Офіс (ofis), стол 120-1965

Table 'Chairs'

id: 1 - Name: Крісло Валенсія CF, WOOD (Valensiya CF, WOOD)-4640
id: 2 - Name: Крісло Ніка CF XROM (Nika CF KHRUM)-4495
id: 3 - Name: Крісло Самба (Samba)-1552
id: 4 - Name: Крісло Urban LB чорний-2819
id: 5 - Name: Крісло Urban HB чорний-3059
id: 6 - Name: Крісло VR Racer BN-W0100-4390
id: 7 - Name: Крісло VR Racer BN-W0110A-4701
id: 8 - Name: Крісло VR Racer BN-W0105A-4390
id: 9 - Name: Крісло VR Racer BN-W0107B-4702
id: 10 - Name: Крісло VR Racer BN-W0109A-4702
id: 11 - Name: Крісло VR Racer Zeus-5544
id: 12 - Name: Крісло Vulcan-4404
id: 13 - Name: Крісло VR Racer Atom-5544
id: 14 - Name: Крісло Urban LB білий-2819
id: 15 - Name: Крісло Urban HB білий-3059
id: 16 - Name: Крісло Madison-3984
id: 17 - Name: Крісло Concept-3059
id: 18 - Name: Крісло SAMBA ULTRA GTP Tilt CHR68 (BOX-2)-6220
id: 19 - Name: Крісло SAMBA ULTRA GTP Tilt CHR68-3110
id: 20 - Name: Крісло SAMBA GTP Tilt CHR10 -2846
id: 21 - Name: Крісло SAMBA GTP Tilt CHR10 (BOX-2)-5692
id: 22 - Name: Крісло SAMBA Ultra T wood chrome (BOX-2)-5568
id: 23 - Name: Крісло SAMBA ULTRA chrome (BOX-2)-5108
id: 24 - Name: Крісло SAMBA T plast chrome (BOX-2)-4816

Додаток 2- файл з функціями для роботи з базою даних

```
<?php

function SelectT($table)

{$servername = "diplomwork-mysqldbserver.mysql.database.azure.com";

$username = "mysqldbuser@diplomwork-mysqldbserver";

    $password = "Alexandr11";

$dbname = "mysqldbdatabase";

// Create connection

$conn = mysqli_connect($servername, $username, $password, $dbname);

// Check connection

if (!$conn) {

    die("Connection failed: " . mysqli_connect_error());

}

$sql = "SELECT MIN(price) as price FROM $table";

$result = mysqli_query($conn, $sql);

$date=mysqli_fetch_array($result);

    echo $date["price"];

mysqli_close($conn);}

function SelectTMAX($table)

{$servername = "diplomwork-mysqldbserver.mysql.database.azure.com";

$username = "mysqldbuser@diplomwork-mysqldbserver";

    $password = "Alexandr11";

$dbname = "mysqldbdatabase";

// Create connection

$conn = mysqli_connect($servername, $username, $password, $dbname);

// Check connection

if (!$conn) {

    die("Connection failed: " . mysqli_connect_error());

}

$sql = "SELECT MAX(price) as price FROM $table";

$result = mysqli_query($conn, $sql);

$date=mysqli_fetch_array($result);

    echo $date["price"];
```

```

mysql_close($conn);}

function Drop_table()
{ $servername = "diplomwork-mysqldbserver.mysql.database.azure.com";
$username = "mysqldbuser@diplomwork-mysqldbserver";
$password = "Alexandr11";
$dbname = "mysqldbdatabase";
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());}
$sql = "DROP TABLE WG_system";
    if($conn->query($sql) === TRUE ) {
        die('Could not delete table: ' . mysql_error());}
    echo "Table deleted successfully\n";
    mysql_close($conn);}

function Cr_table($table)
{ $servername = "diplomwork-mysqldbserver.mysql.database.azure.com";
$username = "mysqldbuser@diplomwork-mysqldbserver";
$password = "Alexandr11";
$dbname = "mysqldbdatabase";
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) { die("Connection failed: " . mysqli_connect_error());
}$sql = "CREATE TABLE $table (
id INTEGER AUTO_INCREMENT PRIMARY KEY,
adress VARCHAR(64),
area VARCHAR(64),
price INTEGER,

```

```

size INTEGER,
id_site INTEGER)";
if ($conn->query($sql) === TRUE) {
    echo "Table created successfully";
} else {
    echo "Error creating table: " . $conn->error;}
mysqli_close($conn);}

function Insert($table,$object)
{
    $servername = "diplomwork-mysqldbserver.mysql.database.azure.com";
    $username = "mysqldbuser@diplomwork-mysqldbserver";
    $password = "Alexandr11";
    $dbname = "mysqldbdatabase";

    // Create connection
    $conn = mysqli_connect($servername, $username, $password, $dbname);

    // Check connection
    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());}

    $columns=array();
    foreach($object as $key=>$value)
    {
        $columns[]=$key;
        $masks[]=$value;
        if($value==null)
        {$object[$key]='NULL';}
        $columns_s=implode(',',$columns);
        $masks_s=implode(',',$masks);
        foreach($masks_s as $value1)
        {echo $value1;}
    }
    $sql="INSERT INTO $table ($columns_s) VALUE ($masks_s)";
    if ($conn->multi_query($sql) === TRUE) {
        echo "New records created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;}
    mysqli_close($conn);}?>

```


Додаток 3-

```
<?php
include('function.php');?>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <title></title>
  <link href="style.css" rel="stylesheet" />

</head>

<body>
  <div class="header">
    <h1>Бізнес Ідея</h1>
    <div class="row">
      <label >
        Шаблони (3)
      </label>
      <form method="post" action="b_function.php">
        <select name="exempl" onchange="this.form.submit()" >
          <option value="-1">—</option>
          <option value="1">
            Дизайн та Реклама
          </option>
          <option value="2">
            IT Компанія
          </option>
          <option value="3">
            Офіс
          </option>
        </select>
      </form>
    </div>
  </div>
  <?php
  $value=3;
  $name='SelectT';
  if(isset($_POST['exempl'])){
    $value = $_POST['exempl'];

    if($value==1){$wg='WG_system';}
    else if($value==2){$wg='WG_system';}
    else{$wg='Office_comp'; }}?>
  <div class="main">
    <table>
    <tr>
      <th>Що купуємо</th>
      <th>Мінімальна вартість</th>
```

```

        <th>Максимальна вартість</th>
</tr>
<tr>
<td>
        <h3>Обладнання</h3>
<tr>
<td>
        Комп'ютери <span class="currency"></span>
</td>
<td>
        <input class="min" name="data[comp]" value="<? $name($wg);?>" type="text">
</td>
<td>
        <input class="max" name="data[comp]" value="<? SelectTMAX($wg);?>" type="text">
</td>
</tr>
<tr>
<td>
        Монітори <span class="currency"></span>
</td>
<td>
        <input class="min" name="data[comp]" value="" type="text">
</td>
<td>
        <input class="max" name="data[comp]" type="text">
</td>
</tr>
<tr>
<td>
        Переферія <span class="currency"></span>
</td>
<td>
        <input class="min" name="data[comp]" value="" type="text">
</td>
<td>
        <input class="max" name="data[comp]" type="text">
</td>
</tr>
</tr>
<tr>
<td>
        <h3> Меблі </h3>
<tr>
<td>
        Стіл <span class="currency"></span>
</td>
<td>
        <input class="min" name="data[furniture]" value="<? $name('Tables');?>" type="text">

```

```

        </td>
    <td>
        <input class="max" name="data[furniture]" value="<? SelectTMAX('Tables');?>" type="text">
    </td>
</tr>
    <tr>
    <td>
        Стілець <span class="currency"></span>
    </td>
    <td>
        <input class="min" name="data[furniture]" value="<? SelectT('Chairs');?>" type="text">
    </td>
    <td>
        <input class="max" name="data[furniture]" value="<? SelectTMAX('Chairs');?>" type="text">
    </td>
</tr>
    <tr>
    <td>
        Шафи <span class="currency"></span>
    </td>
    <td>
        <input class="min" name="data[furniture]" value="" type="text">
    </td>
    <td>
        <input class="max" name="data[furniture]" type="text">
    </td>
</tr>
    </td>
</tr>

    <tr>
    <td>
        <h3> Приміщення </h3>
        <tr>
        <form method="post" name="place" action="b_function.php">
            <td>
                <input type="button" class="action-btn" value="Своє" onchange="this.form.submit()" >
</input>
            </td>
            <td>
                <input type="button" class="action-btn" value="Оренда"
onchange="this.form.submit()"></input>
            </td>
        </tr>
    </td>
</tr>
<?php

```

```

$value=3;
$name='SelectT';
    if(isset($_POST['exempl'])) {
$value = $_POST['exempl'];

    if($value==1){$wg='WG_system';}
    else if($value==2){$wg='WG_system';}
    else{$wg='Office_comp'; }?>
        <tr>
            <td>
                <input class="min" name="data[furniture]" value="<? SelectT('Chairs');?>" type="text">
            </td>
        <td>
                <input class="max" name="data[furniture]" value="<? SelectTMAX('Chairs');?>" type="text">
            </td>
        </tr>
    </form>
</table>
</div>
</body>
</html>

```