

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ**

**Інститут (факультет) Автоматизації і комп'ютерних систем
Кафедра Інформаційних систем**

**«До захисту в ЕК»
Директор інституту(декан факультету)**

(підпис) _____
(прізвище та ініціали)

« ____ » _____ 20__р.

**«До захисту допущено»
Завідувач кафедри**

(підпис) _____
(прізвище та ініціали)

« ____ » _____ 20__р.

**КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА**

Зі спеціальності 122 «комп'ютерні науки»

(код та назва спеціальності)

Освітньо-професійної програми _____ комп'ютерні науки

На тему: Створення інформаційно-облікової системи поселення студентів у гуртожиток НУХТ

Виконав: здобувач 4 курсу, групи КН-4-5 Поляков Микола Олегович
(прізвище та ініціали)

Керівник Мошенский Андрій Олександрович
(прізвище та ініціали)

(підпис)

Консультанти Мошенский Андрій Олександрович

(прізвище та ініціали)

(підпис)

Мошенский Андрій Олександрович

(прізвище та ініціали)

(підпис)

Мошенский Андрій Олександрович

(прізвище та ініціали)

(підпис)

Рецензент Смітюх Ярослав Володимирович

(прізвище та ініціали)

(підпис)

Засвідчую, що в цій кваліфікаційній
роботі немає запозичень із праць
інших авторів без відповідних
посилань.

Здобувач _____
(підпис)

Київ - 2020р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) _____

Кафедра Інформаційних систем

Освітній

ступінь бакалавр

Спеціальність 122 «Комп'ютерні науки»
(код і назва)

Освітньо-професійна програма Комп'ютерні науки

(назва)

ЗАТВЕРДЖУЮ

Завідувач

кафедри _____

“ ____ ” _____ 20 ____ року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Поляков Микола Олегович

(прізвище, ім'я, по батькові)

1. Тема роботи Створення інформаційно-облікової системи поселення
студентів у гуртожиток НУХТ

керівник роботи: Мошенський Андрій Олександрович,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “27” 04 2020 року №269-кс.

Строк подання здобувачем роботи _____

3. Вихідні дані до роботи Сформована угода, облік студентів, web-інтерфейс

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)
Системний аналіз процесу поселення , аналіз аналогів на ринку, логічна та фізична модель бази даних , розробка бази даних , створення інтерфейсу програми, інструкція користувача , вибір елементної бази, web-інтерфейс.

5. Перелік графічного матеріалу

Організаційна структура університету, Організаційна структура процесу поселення, логічна та фізична моделі бази даних інтерфейс програми web-інтерфейс

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Вступ			
I	Мошенський А.О., Доцент		
II	Мошенський А.О., Доцент		
III	Мошенський А.О., Доцент		
Висновок			

7. Дата видачі завдання _____ 13 січня 2020 року _____

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Передпроектне дослідження та системний аналіз процесу поселення студентів в гуртожиток	27.04.2020	Виконано
2.	Розробка функціональної моделі процесу поселення студентів в гуртожиток	04.05.2020	Виконано
3.	Розробка моделей та створення бази даних	08.05.2020	Виконано
4.	Визначення та реалізація функцій системи	18.05.2020	Виконано
5.	Розробка заходів з охорони праці	22.05.2020	Виконано
6.	Оформлення пояснювальної записки	25.05.2020	Виконано
7.	Розробка презентації	27.05.2020	Виконано

Здобувач _____
(підпис)

Поляков Микола Олегович _____
(прізвище та ініціали)

Керівник роботи _____
(підпис)

Мошенський Андрій Олександрович _____
(прізвище та ініціали)

АНОТАЦІЯ

У вік технологій, звичайні контролі пропуску потребують оновлень. Що є одним з пунктів безпека життя людей, незважаючи на тип будівлі. Поширення цієї проблеми є поштовхом для прогресу в цій конкретній області.

Впровадження електронної системи моніторингу та контролю є засобом яка має полегшити процес поселення в гуртожиток студентів Національного Університету Харчових Технологій. Зменшити кількість пограбування, поганих вчинків і т.п.

Система контролю доступу – це сукупність програмно-технічних засобів та організаційно-методичних заходів, за допомогою яких вирішується завдання контролю і управління відвідуванням окремих приміщень, а також оперативний контроль переміщень на території об'єкта.

Мета дослідження: створення системи обліку проходження студентів по унікальному значенню-ключ.

Завдання дослідження: вибір елементної бази та розробка програмного забезпечення для збору даних проходження студентів крізь систему контролю.

КЛЮЧОВІ СЛОВА: ЕЛЕКТРОННА СИСТЕМА, МОНІТОРИНГ, БАЗА ДАНИХ, WEB-СТОРІНКА, МІКРОКОНТРОЛЕР, ESP8266, RDM6300.

ANNOTATION

In the age of technology, conventional pass controls need updating. Which is one of the points of safety of life of people, regardless of the type of building. The spread of this problem is an impetus for progress in this particular area.

The introduction of an electronic monitoring and control system is a tool that should facilitate the process of settling students in the dormitory of the National University of Food Technology. Reduce the number of robberies, bad deeds, etc.

The access control system is a set of software and hardware and organizational and methodological measures, which solve the problem of control and management of visits to individual premises, as well as operational control of movements on the site.

The purpose of the study: to create a system of accounting for the passage of students on a unique value-key.

Objectives of the study: the selection of the element base and the development of software for collecting data on the passage of students through the control system.

KEY WORDS: ELECTRONIC SYSTEM, MONITORING, DATABASE, WEB PAGE, MICROCONTROLLER, ESP8266, RDM 6300.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ДЛЯ СИСТЕМИ ОБЛІКУ СТУДЕНТІВ В ГУРТОЖИТКУ	10
1.1 Характеристика інформаційно-облікової системи поселення студентів у гуртожиток НУХТ	10
1.2 Організаційна структура поселення студентів в гуртожиток НУХТ	11
1.2.2 Дослідження організації поселення	12
1.2.3. Взаємодія підрозділів та відділів які приймають участь в поселенні студентів в гуртожиток НУХТ	14
1.3. Аналіз нинішнього стану автоматизації	16
1.4 Аналіз, та методи вирішення	17
1.5 Аналіз існуючих систем обліку	19
1.5.1 Система обліку компанії Антарес-2000,2 технічні характеристики	20
1.5.2 Система обліку компанії SSBB	21
1.5.3 Система обліку компанії ZKSoftware	24
1.5.4 Система обліку компанії AxxonSoft	27
1.5.5 Порівняння систем-аналогів	30
1.6 Обґрунтування доцільності розроблення й впровадження електронної системи обліку	32
1.7 Оновлена модель інформаційно-облікової системи	33
1.8 Постановка задачі	35
1.8.1 Призначення та цілі створення системи	35
1.8.2 Вимоги до дипломного проекту	35
1.8.3 Користувачі системи	39
1.8.4 Вибір операційної системи	40
1.8.5 Вимоги до програмного забезпечення	40
Інтерфейс користувача	41
Збереження даних	41
Надійність	41
1.8.6 Функції системи	41
1.8.7 Вхідні , та вихідні данні	42
РОЗДІЛ 2. Створення інформаційно-облікової системи поселення студентів у гуртожиток НУХТ	43
2.1 Розробка макета системи	43
Функціональні модель управління проектом «AS-IS»	43
Інтерфейс користувача	46
Інструкція користувача	47
2.3 Вибір елементної бази	50
2.3.1 Вибір мікроконтролера	50

2.3.3 Вибір інших комплектуючих	56
Огляд середовища Arduino IDE для реалізації програмної частини	59
Web інтерфейс	60
Розділ 3 Охорона праці	61
Система управління охороною праці	61
1. Загальні положення	61
ВИСНОВОК.....	69
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	70
ДОДАТОК «А».....	72
ДОДАТОК «В»	74
ДОДАТОК «С»	78
ДОДАТОК «D».....	81
ДОДАТОК «G».....	82
ДОДАТОК «K».....	84

ВСТУП

Створення електронної системи, ведення обліку відвідування, є актуальним завданням для гуртожитків закладів освіти, та будівель типу гуртожиток.

Для вирішення використовуються різноманітні комплекси обліку. До складу цих комплексів входять засоби реєстрації, засоби збереження, засоби оброблення даних та іноді засоби керування вторинними параметрами.

В наш час існує багато видів систем контролю, але не всі можуть зробити конкуренцію більш авторським, програмованим, виробам. Багато з них мають дуже маленький набір функцій. Із за цього виникло питання, достатньо цих наборів функцій на деяких підприємствах або в жилих домах? Порівнявши компанії які виготовляють це обладнання та надають деякий спектр послуг можна стверджувати що функцій може і не вистачати. Тому мій дипломний проект направлений на вдосконалення, розширення можливостей вже існуючих систем контролю для будівель типу гуртожиток.

Одним з наявних шляхів підвищення безпеки людей та збереження їх речей під більшим контролем є формування обліку входу та виходу з будівлі по власному унікальному ключу, прив'язка кімнати гуртожитку до перепустки. Завдяки цьому буде покращена ефективність безпеки людей в будівлі. Є можливість використання програми для будь яких систем контролю, тому проект є унікальним.

Мета дослідження: створення інформаційно-облікової системи обліку даних

Завдання дослідження: вибір елементної бази та розробка програмного забезпечення для обліку проходження студентів через системи контролю.

Об'єктом дослідження є облік проходження студентів через систему контролю неперервної дії

Предмет дослідження є процес збору даних по даті, та часу проходження людей через систему контролю та ведення обліку.

Методологічно-теоретичною основою є теорія обробки інформації, теорія кодування, основи електротехніки та електроніки, теоретичні основи систем зв'язку, без проводів системи передачі даних.

Наукова новизна розроблено та впроваджено подальшого розвитку система обліку проходження студентів, працівників до гуртожитку. Надання швидкої інформації про місце поселеного студента.

Практична цінність – зменшення затрат часу на формування документів при поселенні, формування звітності в електронному варіанті, а не паперовому, швидкий пошук про поселеного студента або робітника вищого навчального закладу.

Облік часу та ідентифікування системою особи яка проходить по особливій картці. Можливість життєздатності програми/бази у разі знеструмлення, а саме швидке перезавантаження системи від акумуляторів або від універсальних мобільних пристроїв.

Завданням роботи є вибір елементної бази, а саме: мікроконтролерів, засобів зв'язку, написання мікро програмного коду для частини обліку даних, та серверної частини зберігання та оброблення даних.

РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ДЛЯ СИСТЕМИ ОБЛІКУ СТУДЕНТІВ В ГУРТОЖИТКУ

1.1 Характеристика інформаційно-облікової системи поселення студентів у гуртожиток НУХТ

Актуальність цієї роботи полягає у тому що потрібно модернізувати, оновити системи, які є застарілими в наші часи, це стосується і принципів та методів. Не зважаючи на те що ми зараз живемо в часи індустрії, все одно більшість процесів з якими ми пов'язані, ми робимо за застарілими стандартами. Описати коротко, як приклад, процес поселення. Чому заводи, фабрики, якісь підприємства таким чи іншим чином автоматизують щось, а такі процеси ні? Тому що перекладання паперової документації не така й складна робота щоб її автоматизувати. Тому мій проект спрямований не скільки полегшити, а прискорити процес поселення студентів в гуртожиток НУХТ.

Зараз щоб поселитись у гуртожиток потрібно багато бігати с паперовою документацією. Перші курси які поселяються в гуртожиток вони мають відстояти чергу декілька годин.

Мета цього проекту, зменшити затрати часу, натовп людей та максимально зменшити кількість паперу в формуванні звітів, шахматках розселення тощо. Система яка буде вести облік проходження студентів крізь контроль пропуску в гуртожиток.

1.2 Організаційна структура поселення студентів в гуртожиток НУХТ

Основними складовими функціями процесу поселення в гуртожиток НУХТ є формування звітів, угод, ордерів, обробка документів студентів, формування списків, наказ на розподіл кімнат між студентами, формування «шахматки» житлового фонду, формування наказу поселення в гуртожиток.

Процес поселення студентів в гуртожиток забирає багато часу. Починаючи зі збору інформації зарахованих студентів до формування та підписання наказу. Виникла ідея по спрощенню процедури оформлення документів необхідних для поселення в гуртожиток. Швидке формування звітів, ведення обліку студентів в гуртожитку, реєстрація пропусків. Завдяки цій моделі буде зменшено затрати часу та відбудеться суттєве підвищення ефективності поселення, спроститься ведення обліку студентів.

Формування ордерів та прив'язка їх до ПІБ студента і дати народження. Закріплення студента за конкретною кімнатою. Директор веде облік порожніх та зайнятих кімнат. Відображення кількості студентів в одній кімнаті. Інформація по оплаті за проживання в гуртожитку НУХТ. Буде створено базу даних в якій буде занесена інформація про студента , закріплений пропуск з унікальним ключем, ордер, номер кімнати, графа за порушення внутрішнього розпорядку гуртожитка. Формування електронних заяв для поселення студентів до гуртожитку НУХТ.

Можливість додавати в базу студента університету у будь який час після фактичного поселення. Зменшити кількість паперового документообігу , а краще замінити його на електронний облік поселених студентів.

На рис. 1.1. зображено верхній рівень організаційної структури університету.

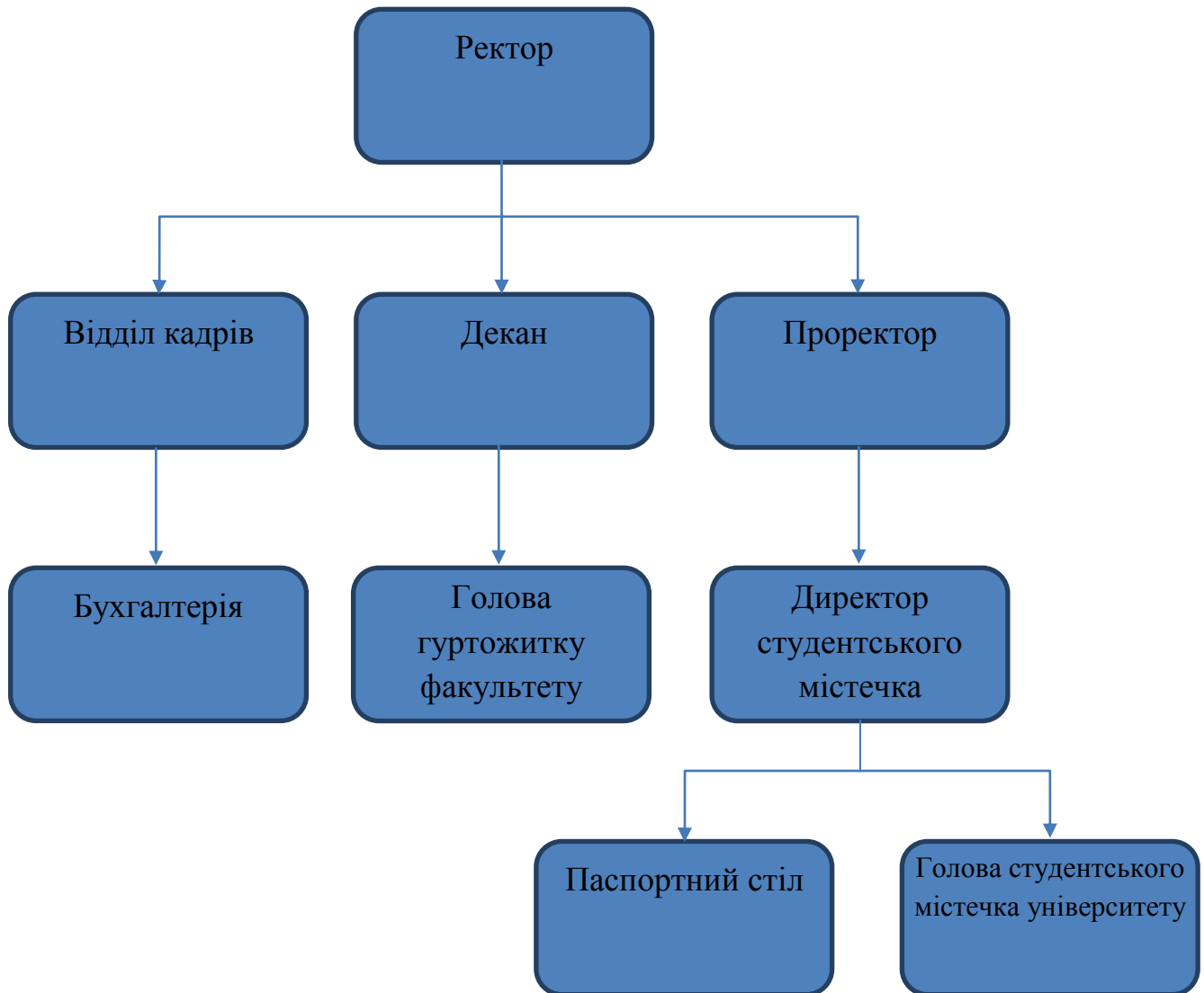


Рис.1.1. Організаційна структура університету

1.2.2 Дослідження організації поселення

В поселенні студентів приймає участь Директор студентського містечка, голова студентського містечка університету, голова гуртожитку факультету, комендант та нічний комендант.

Організаційна структура має лінійний вигляд. Це свідчить про те, що планування робіт і контроль їх виконання здійснюється від директора студентського містечка.

Директор студентського містечка формує наказ на розподіл кімнат між факультетами. Коменданту гуртожитку передається список зарахованих людей, які потребують гуртожиток і він передає цей список голові гуртожитку. Голова гуртожитку разом із головою студентського містечка формує шахматку розселення студентів.

Після сформованої «шахматки» розписуються номери ордерів для студентів. Ордер має бути унікальним для кожного студента. Ордер це номер за яким студент закріплений по номеру кімнати.

На рисунку 1.2. зображено організаційну структуру процесу поселення.

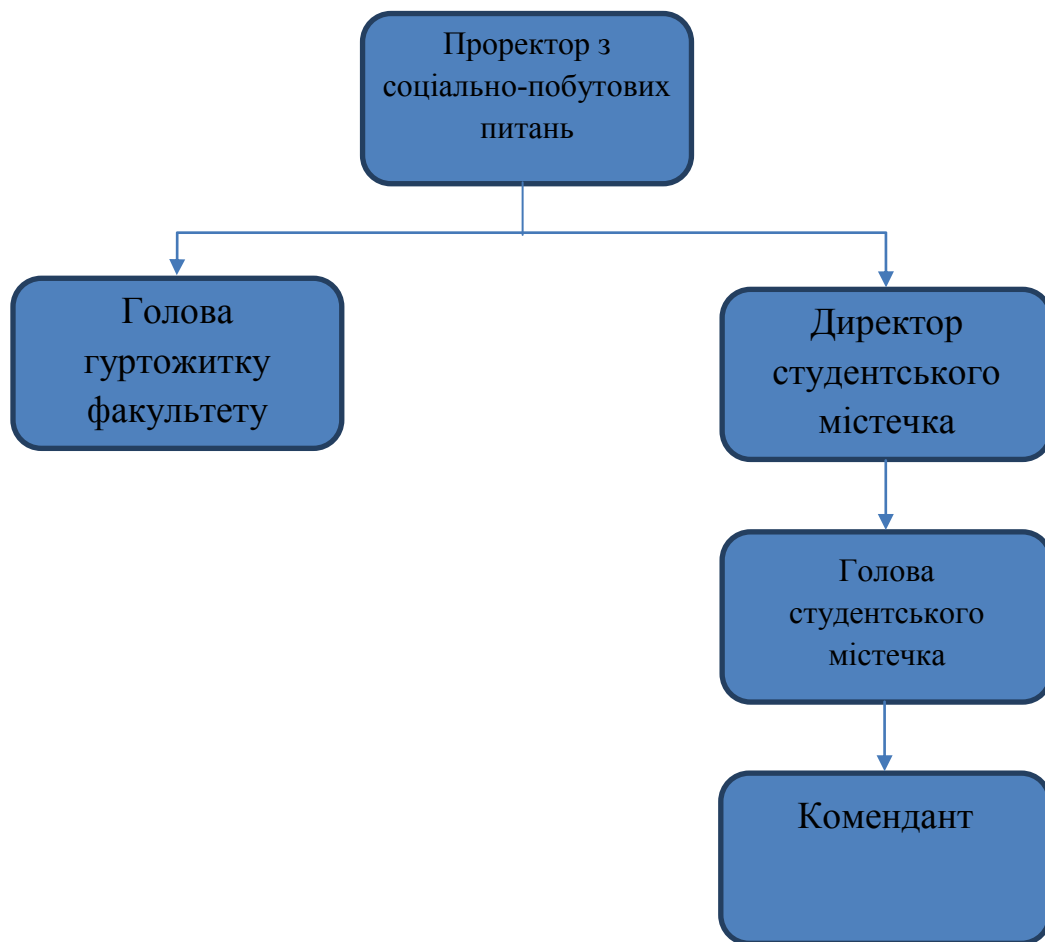


Рис.1.2. Організаційна структура процесу поселення

1.2.3. Взаємодія підрозділів та відділів які приймають участь в поселенні студентів в гуртожиток НУХТ.

Задачі, які потребують автоматизації

Покладаючись на високу інтенсивність і складність бізнес-процесів, що відбуваються в підрозділі, система обліку та управління виробництвом повинна надавати єдиний інформаційний простір, що забезпечує ефективну регламентовану взаємодію підрозділів та відділів університету НУХТ.

Для цього необхідно автоматизувати наступні процеси:

- створення швидкого та зручного формування ордерів та угод
- організувати швидкий доступ до бази в яку буде заноситись номери кімнат та ордерів
- можливість додавати в базу у будь який момент звітів про виконання поселення
- формування наказу на поселення.

Система повинна мати функціонал, що надає більш детальну інформацію в першу чергу для коменданта, вахтера, а саме: номер кімнати студента, оплату за гуртожиток, облік студента коли зайшов , вийшов.

№	Підрозділ	Одержання	Надання
1	З приймальною комісією	– Списки зарахованих студентів які потребують гуртожиток;	– відомостей про розподіл кімнат;
2	З директором студентського містечка	– наказ на розподіл кімнат між інститутами та факультетами;	– даних про кількість зарахованих студентів , які потребують гуртожиток ;
3	З головою гуртожитку	– Списки студентів на поселення;	– Формування шахматки;

№	Підрозділ	Одержання	Надання
4	З бухгалтерією	– підсумки інвентаризації; – звіти про стан гуртожитку; – документи, накази, розпорядження, необхідні для здійснення бухгалтерського обліку і контролю.	– Різного роду довідки, відомості та інші документи.

Таблиця 1.1. Взаємодія підрозділів та відділів які приймають участь в поселенні студентів в гуртожиток НУХТ.

1.3. Аналіз нинішнього стану автоматизації

Зараз в гуртожитку взагалі немає автоматизованих систем. Вся документація для поселення зберігається в паперовому виді. Студент пише заяву на аркуші А4 від руки. Всі ці заяви відносяться до голови гуртожитку і він в свою чергу формує таблицю розселення студентів в програмному засобі Microsoft Office Word.

1.4 Аналіз, та методи вирішення

Процес автоматизації є дуже складним з точки зору аналізу, та виявлення проблеми на підприємстві, тощо.

Автоматизація — є одним з напрямів науково-технічного прогресу, який спрямовано на застосування саморегульованих технічних засобів, економіко-математичних методів і систем керування, що звільняють людину від участі у процесах отримання, перетворення, передавання і використання енергії, матеріалів чи інформації, істотно зменшують міру цієї участі чи трудомісткість виконуваних операцій. Разом з терміном автоматичний, використовується поняття автоматизований, що підкреслює відносно великий ступінь участі людини у процесі.

Поміняти повністю процес поселення буде майже не можливо оскільки треба починати міняти всі відділи в Університеті. Зменшити затрати часу в моменти поселення та зробити облік проходження все ж таки можливо і є доцільним проектом.

Наприклад формування заяв на поселення робляться в ручну. Можна зробити та обробляти в електронному вигляді таблиці розселення.

Система обліку проходження студентів через контроль пропуску в гуртожитку НУХТ відсутня. Використання такої системи є доцільним. Такий проект підніме безпечність перебування в гуртожитку.

USED AT:	AUTHOR: nicholas	DATE: 21.11.2019	WORKING	READER	DATE	CONTEXT:
	PROJECT: спростити поселення в гуртожиток студентів	REV: 17.05.2020	DRAFT			TOP
			RECOMMENDED			
	NOTES: 1 2 3 4 5 6 7 8 9 10		PUBLICATION			

Житловий фонд

Документи студентів

Наказ про зарахування

Нормативно правова документація

Накази і розпорядження

Статут гуртожитку

Створення інформаційно-облікової системи поселення студентів у гуртожиток НУХТ

База розселення студентів

Наказ про поселення в гуртожиток

Приймальна комісія

Ректор

Голова студентського містечка

Голова студентської ради гуртожитку

0

NODE:	TITLE:	NUMBER:
A-0	Створення інформаційно-облікової системи поселення студентів у гуртожиток НУХТ	

1.5 Аналіз існуючих систем обліку

Для кращого розуміння принципу роботи та можливостей систем обліку розглянемо існуючі системи на ринку, а саме системи:

- Антарес-2000
- SSBB
- ZKSoftware
- AxxonSoft

1.5.1 Система обліку компанії Антарес-2000,2 технічні характеристики

Антарес-2000- Система контролю доступу та обліку робочого часу (СКД, СКУД) являє собою програмно-апаратний комплекс, який вирішує завдання з управління доступом на територію об'єкта, підприємства, офісу, або локального обмеження доступу співробітників в різні зони, а також управління виконавчими пристроями.

Можливості пристрою і програми :

1. Перш за все, система контролю доступу (СКД, СКУД) дозволяє ефективно припиняти прохід до будівлі особам, які не мають на це право, а так само вводити і оперативно змінювати рівень доступу працівників організації в різні сектори і приміщення.
2. Система контролю доступу дозволяє вести облік статистики проходів через кожен точку доступу зі збереженням інформації про час проходів і про особу. Це дозволяє формувати звіт про дотримання співробітниками графіка робочого часу з фіксацією запізнень на роботу або передчасного відходу з неї, наявності прогулів і часу перебування поза будинком у службових завданнях в робочий час.
3. За допомогою СКД легше вирішувати питання безпеки в будівлі, система фіксує спроби несанкціонованого доступу і повідомляє про них на пульт охорони.

1.5.2 Система обліку компанії SSBB

Компанія «Сучасні системи безпеки бізнесу» надає повний спектр послуг зі створення комплексної системи безпеки на об'єкті. В тому числі, їх фахівці мають величезний досвід в проектування, встановлення та налаштування систем обліку робочого часу.

Ця система може контролювати місце розташування співробітників, складати детальний звіт по робочому часу співробітника за певний період часу (день, тиждень, місяць). Система складається з сервера зі спеціальним програмним забезпеченням, контролера і зчитувача. За зчитувача розрізняють види систем контролю, вони бувають карткові, брелочная і біометричні.

Складові системи обліку від компанії SSBB

- Програма обліку робочого часу
- Хронометраж робочого часу
- Графік обліку робочого часу
- Системи контролю доступу та обліку робочого часу
- Термінал обліку робочого часу
- Біометрична система обліку робочого часу
- Підсумований облік робочого часу

Програма обліку робочого часу:

Це один з основних елементів системи УРВ, ця програма встановлюється на сервер. Принцип роботи програми обліку робочого часу складається в періодичному «опитуванні» (автоматично або в ручному режимі) контролерів системи і зчитування подій з них, систематизації інформації та надання даних користувачеві.

Хронометраж робочого часу

Це вид звіту системи УРВ в якому детально розписані всі пересування персоналу по точках проходу протягом певного проміжку часу. Хронометраж робочого часу може складатися за конкретний день, тиждень, місяць або певний проміжок часу.

Графік обліку робочого часу

Можна ще назвати графіком робочого часу, це функція яка дозволяє контролювати запізнення і ранні відходи персоналу. Графік УРВ спочатку визначається в налаштуваннях програми і далі при формуванні звіту по робочому часу програма сама звертає увагу користувача на ранні відходи і запізнення персоналу.

Системи контролю доступу та обліку робочого часу

У комплексі уявляю собою систему безпеки закритій території або приміщення. СКД і Сурва надає функції ефективного дистанційного керування доступом персоналу до об'єктів, а також контролю за місцезнаходженням і можливістю обліку робочого часу персоналу.

Ця система є невід'ємною частиною комплексної системи безпеки на об'єкті, що охороняється поряд з відеоспостереженням і охороною.

Термінал обліку робочого часу

Складається з двох зчитувачів (вхід, вихід) контролера і замку. Термінал УРВ встановлюється на точку проходу, збирає інформації про що проходить персонал і передає її на сервер. Контролер також управляє замком на точці проходу за певними правилами, отриманим з сервера. сучасні термінали обліку робочого часу також комплектуються доводчиком і датчиком відкриття дверей для надання повної інформації про стан точки проходу.

Біометрична система обліку робочого часу

Це сучасна система обліку робочого часу, яка працює на основі сканування та аналізу однієї з біометричних характеристик людини. Біометрична система обліку робочого часу дозволяє максимально

виключити несанкціонований доступ, передачу ключа, несанкціоновану позначку про початок робочого дня або його закінчення.

На сьогоднішній день біометричні системи обліку робочого часу можуть працювати з відбитками пальців (найдоступніший і найпоширеніший варіант), сканування сітківки ока, аналіз фото особи, аналіз голосу. Недоліком біометричних систем є так звана «несработка». Справа в тому, що при скануванні людина може відчувати певний дискомфорт (навіть страх), при цьому з організмом відбуваються деякі зміни (зволоження рук або очей, часте моргання, тремтіння рук і т.д.) і, відповідно, системі необхідно повторне проходження процедури ідентифікації. Згодом персонал звикає проходження процедури і кількість «несработок» зменшується.

Підсумований облік робочого часу

Так само підсумковий облік робочого часу звіт по робочому часу сформований за якийсь регламентований (звітний) період часу (тиждень, місяць, квартал, рік). Підсумований облік робочого часу (підсумковий облік робочого часу) можна узгодити на програмному рівні з бухгалтерською програмою та нараховувати за допомогою нього заробітну плату співробітникам в меню програми, в якому, задаючи вихідні характеристики (конкретний персонал, місце, час), можна отримати місце розташування зазначеного персоналу.

1.5.3 Система обліку компанії ZKSoftware

Біометрична система обліку робочого часу zksoftware

Тип обладнання - термінал, облік робочого часу

Термінали ZKTeco- це об'єктивний облік робочого часу співробітників. Достовірність обліку гарантується тим, що ідентифікатором користувача є відбиток пальця, який не можна забути, втратити, передати іншому для фіксації часу присутності без власника ідентифікатора. Тільки особиста присутність співробітника необхідно для фіксації свого приходу на робоче місце. В кінці облікового періоду Вам залишиться тільки сформувати табель робочого часу одним натисненням кнопки, що дозволить максимально мінімізувати час роботи бухгалтера на ведення ручних записів для складання табеля відвідуваності.

Зовнішній вигляд

Термінал біометричний СУРВ - це невеликий мікрокомп'ютер з вбудованим екраном і клавіатурою. Для ідентифікації користувача по відбитку пальця використовується вбудований оптичний сенсор сканування відбитків, за безконтактною карті- вбудований RFID модуль карт Em-Marine.

На бічній стороні терміналу є Usb слот для підключення USB диска (завантаження / розвантаження подій відвідуваності і користувачів) і кнопка Reset для перезавантаження обладнання.

На задній стороні знаходяться роз'єми для підключення елементів системи СКУД і мережевий інтерфейс Ethernet.

Вбудована резервна батарея забезпечує живлення обладнання обліку протягом 4 годин при відсутності основного живлення.

Біометричні характеристики

3000 шаблонів відбитків. Швидкість розпізнавання відбитків не більше 2с. Високоякісний оптичний сенсор ZK для точної і швидкої ідентифікації.

Програмування пристрою

Для автономного програмування терміналу використовується Меню пристрою. Для цього натискається кнопка Меню, потім Реєстрація нового користувача. Далі користувач прикладає 3 рази один і той же палець до сенсора.

При підключенні U160-C в локальну комп'ютерну мережу, можливо створити базу шаблонів відбитків за допомогою настільного USB сканера ZK4000, і вивантажити відбитки в пам'ять пристрою.

Інтерфейси зв'язку з ПК

- Ethernet- мережевий інтерфейс для організації системи безпеки по відбитку пальця при великій кількості точок проходу.
- Usb host- ручне скачування подій відвідуваності на Usb диск для подальшого перегляду на комп'ютері.
- Usb client- пряме підключення пристрою по usb інтерфейсу до комп'ютера.

Монтаж обладнання

Термінал доступу - це швидке і легке розгортання системи. У комплекті з устаткуванням поставляється пластина для кріплення на стіну. Або встановлюється на столі біля секретаря.

Технічні характеристики:

Кількість відбитків	3000
Кількість записів	100 000
Сенсор ZK	оптичний

Версія алгоритму	ZK Finger v10.0
Вбудований RFID рідер	125kHz RFID Em-Marine
Інтерфейси зв'язку	RS485, TCP / IP, USB-host
Живлення	5В DC 500 мА
Робоча температура	0 - + 45 градусів Цельсія
Вологість	10% -90%
Розміри	192x140x43 мм

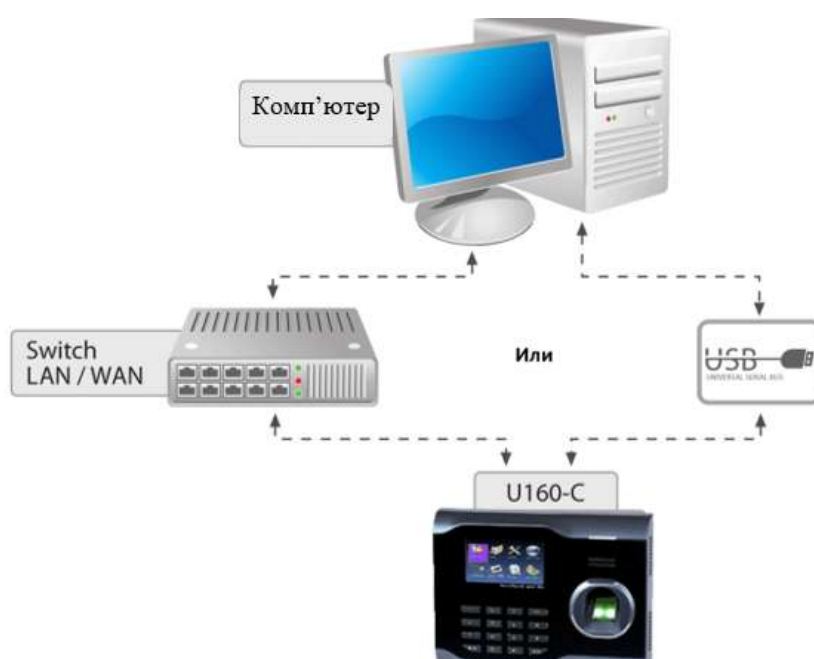


Рис1.3 схема з'єднання

1.5.4 Система обліку компанії AxxonSoft

Автоматизація підрахунку робочого часу

Модуль обліку робочого часу дозволяє формувати графіки і схеми роботи, призначати робочі приміщення для кожного співробітника і відділу в цілому. Використовуючи дані системи контролю доступу про переміщення співробітників, модуль розраховує відпрацьований час для кожної людини і всього відділу з урахуванням виправдувальних і понаднормових документів, а потім представляє результат у вигляді таблиці або графіка.

Модуль УРВ підтримує різні графіки:

- тижневий (робочу п'ятиденку);
- змінний (наприклад, доба через дві);
- місячний (наприклад, коли робочими є парні дні місяця);
- ковзає: співробітник повинен відпрацювати декілька годин в певний час (наприклад, 8 годин на інтервалі з 8:00 до 22:00).

За допомогою модуля УРВ можна:

переглядати кадрову структуру підприємства по відділах і отримувати інформацію по кожному співробітнику (кадрова структура створюється в «Інтелект»);

отримати відомості про час присутності співробітника на робочому місці, часу його роботи в нічні години, межею впливу часу, переробці;

автоматизувати виявлення порушень трудового розпорядку і отримати звіт щодо порушень за певний період часу;

скласти звіт по сумарному відпрацьованому часу співробітників підрозділів в Уніфікованій формі №Т-12 / Т-13, затвердженій наказом Міністерства України, з можливістю друку і подальшого редагування;

- Зберігати звіти в форматах doc, xls, txt, bmp, jpeg, tiff і gif.

- Побудова звітів

Дані, одержувані від модуля контролю доступу, використовуються для складання звітів за допомогою підсистеми Intellect Web Report System.

У будь-який момент за будь-який звітний період можна отримати відомості у вигляді статистики про відпрацьовані кожним співробітником годин (з урахуванням часу, відпрацьованого в нічні години, всіх запізень, відряджень відпусток, відгулів і лікарняних).

Report System

3.0.0.0.0

General reports Time and Attendance reports

Navigation: Time and Attendance reports > Employee time clock report > Result

Page 1 Item 1 PDF 100%

Arrival-Leaving report

Period: 08.12.2014 - 14.12.2014

Department Axoon

Personnel number	Full Name	Position	Time of arrival/leaving							
			Time of presence				Hours worked			
			08.12.2014	09.12.2014	10.12.2014	11.12.2014	12.12.2014	13.12.2014	14.12.2014	
	Alex Gernas	Tech writer				09:10 / 18:45				
1	Eugene Pak	Support Engineer				09:11 / 18:48				
	Ratin Jacobson	Tech consulting				09:35 / 9:35				
	Kulova Anastasia	Tech writer								
	Lisa Jefferson	Tech writer				09:10 / 18:45				
	Vorobyova Olga	Tech writer				09:20 / 18:53				

Рис.2.1. форма звітності по кількості днів на роботі

Firefox * Report System

localhost:8081/reports/WorkTime/Report/ReportResult

Report System
v.1.0.222

General reports Time and Attendance reports

Navigation: Time and Attendance reports > Hours-worked report

Page 1 from 1 PDF 100%

Date: from 11.12.2014 to 11.12.2014

Hours worked

Department: Axxon

Full Name	Time of presence	Working time	Time of absence	Night time	Overtime
Alex German axxon.work.schedule	00:35:00	07:49:10	00:10:50	00:00:00	00:45:50
Eugene Pak axxon.work.schedule	00:35:01	07:48:11	00:11:49	00:00:00	00:46:50
Katrin Jacobson axxon.work.schedule	00:00:00	00:00:00	08:00:00	00:00:00	00:00:00
Kirilova Anastasia axxon.work.schedule	00:00:00	00:00:00	08:00:00	00:00:00	00:00:00
Lisa Jefferson axxon.work.schedule	00:35:00	07:49:33	00:10:27	00:00:00	00:45:27
Vorobyova Olga axxon.work.schedule	00:32:40	07:39:30	00:20:30	00:00:00	00:53:18
Total by department:	38:17:49	31:06:24	16:53:36	00:00:00	03:11:25

Total by report

Total by report:	38:17:49	31:06:24	16:53:36	00:00:00	03:11:25
-------------------------	-----------------	-----------------	-----------------	-----------------	-----------------

Рис.1.4 Форма звітності по кількості годин на роботі

1.5.5 Порівняння систем-аналогів

Системи-аналоги порівнюються за такими характеристиками:

Масштабованість – можливість додавання нових функцій в ІС або зміни деяких існуючих функцій та модулів, при цьому решта функціональних частин ІС мають залишатися незмінними

Рівень складності масштабованості необхідний рівень знань для розширення системи

Мобільність – можливість перенесення програм і даних при модернізації або заміні апаратних платформ ІС і можливість роботи після цього користувачів без їх перепідготовки

Розподіленість – можливість роботи з системою віддалено через мережу Інтернет.

Можлива децентралізація здатність системи працювати без централізованого вузла, або мати декілька централізованих вузлів

Наявність Бази даних забезпечує можливість зберігання даних у БД в електронному вигляді.

Автоматичне формування звітів формування звітів за період .

Наявність довідкової системи дозволяє користувачу програми без сторонньої допомоги використовувати функції ІС.

Захист доступу до даних необхідно ввести логін та пароль для входу в систему

Ціна продукту.

Операційні системи, що підтримують додаток.

Результати порівняння систем-аналогів представлені в табл. 1.1.

Таблиця 1.1 Порівняння систем-аналогів

Характеристика	Антарес-2000	SSBB	ZKSoftware	AxxonSoft
Масштабованість	+-	+-	+-	+
Рівень складності масштабованості	низький	середній	середній	середній
Мобільність	+	+	-	-
Розподіленість	+	+	+	+
Можлива децентралізація	-	-	+	-
Наявність Баз даних	+	+	+	+
Автоматичне формування звітів	-	-	+	+
Наявність довідкової системи	+	+	+	+
Підтримка з боку розробника	+	+	+	-
Захист доступу до даних	+	+	+	+
Операційні системи	Linux, Windows	Windows	Windows, Linux, Mac OS	Windows, Linux, Mac OS
Ціна продукту	9200 грн.	11110 грн.	14000 грн.	8700 грн.

З приведених систем найкраще себе показала система ZKSoftware.

Її недоліками є ціна та складність у масштабованості. У результаті аналізу вищерозглянутих систем приходимо до висновку, що ці системи не варто впроваджувати у гуртожитках НУХТ або будинках типу гуртожиток .

1.6 Обґрунтування доцільності розроблення й впровадження електронної системи обліку

Система має місце існування тому що завдяки цій системі зменшиться кількість пограбувань , порушень, тощо. Завдяки цій системі можна зменшити коло потенційних правопорушників тим самим облегшити пошук.

Спроститься ведення обліку поселених студентів гуртожитку, прискориться пошук розміщеного студента. Буде зберігатися база студентів, які колись проживали, звітність за минулі роки, всі порушення тощо.

Зберігання документації в електронному вигляді по всіх параметрах краще ніж в паперовому.

Порівнявши системи обліку, створивши таблицю, виникла проблема, що унікального рішення немає. Як було зазначено в пункті 1.3 проблеми які є зараз, потрібно вирішити. А саме відцифрувати паперові документи в електронний варіант. З наведених прикладів, опис яких є в пункті 1.5, було прийнято рішення вдосконалити вже існуючі системи обліку. Вдосконалення систем різними функціями яких не було до цього.

Тому мій проект спрямований вирішити проблему с затратами часу при поселені, та створення інформаційно-облікової системи, яка була б дешевшою від своїх систем-аналогів, та мала більший спектр функцій.

1.7 Оновлена модель інформаційно-облікової системи

Модель TO-BE

Модель TO-BE («як повинно бути») створюється на основі AS-IS, з усуненням недоліків в існуючій організації бізнес-процесів, а так само з їх удосконаленням та оптимізацією шляхом усунення виявлених на основі аналізу AS-IS вузких місць.

Відповідно до моделі TO-BE метою предмета розробки є спрощення організації обліку і руху товарів на фірмі. При цьому предмет розробки повинен забезпечити:

- формування бази даних, інформацію про студентів;
- засоби доступу до цієї бази для редагування, додавання, видалення даних;
- сортування і фільтрацію даних, що містяться в базі;
- засоби пошуку потрібної інформації в базі даних;
- зручний перегляд запитаної інформації.

На основі аналізу створеної вище AS-IS моделі процесів проблемної області була створена TO-BE модель, контекстна діаграма якої наводиться на малюнку рис.1.2

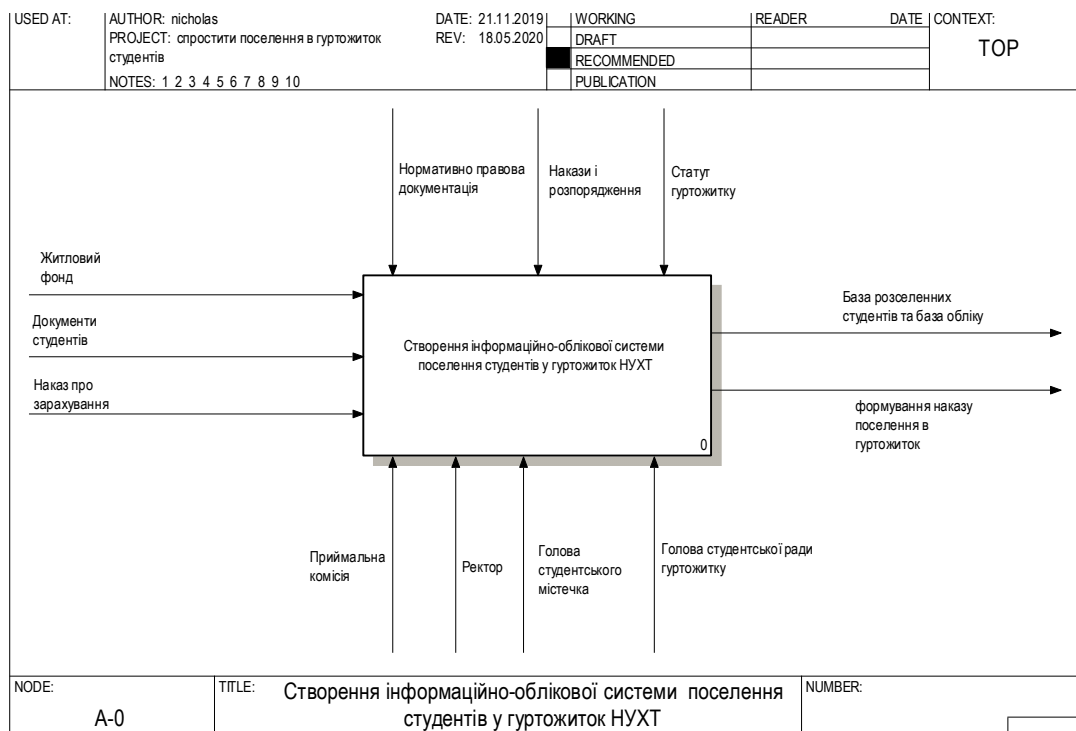


Рис.1.4 Модель TO-BE

Взявши за основу попередню модель AS-IS, для створення моделі TO-BE, було внесено зміни в створенні бази не тільки розселення а й створення бази обліку студентів , для можливості слідкування проходження студентів, зміни в формуванні ордерів , тощо.

1.8 Постановка задачі

1.8.1 Призначення та цілі створення системи

Враховуючи все вище сказане у пункти 1.7., система має зменшити затрати часу та відцифрувати паперові документи на електронні носії. Пошук важливої інформації прискориться, і буде зменшено кількість паперового документообігу.

Система створюється для гуртожитку НУХТ, але вона є унікальною для всіх будинків типу гуртожиток. В даному проекті користувачем виступає голова гуртожитку, або комендант в залежності від задачі яку потрібно виконати.

Голова гуртожитку факультету поселяє та вносить студентів в базу даних.

Комендант займається пошуком поселених студентів при необхідності.

1.8.2 Вимоги до дипломного проекту

Цей дипломний проект є не сильно ресурсо-підйомний вистачить звичайного комп'ютера та мікроконтролер, та комплектуючі.

Вимоги щодо технічного (апаратного) забезпечення:

Процессор					
Модель	Ціна	Частота	Сокет	Кэш	Потужність
Intel Xeon X3440	\$7,1	2,53 ГГц	LGA 1156	8 МБ	95 Вт
INTEL XEON E5430	\$8.25	2.6 ГГц	LGA 771	12 МБ	80 Вт
Intel Xeon L5420	\$7.08	2.5 ГГц	LGA 771	12 МБ	50 Вт
Intel Xeon e5-2620	\$7.86	2.0 ГГц	LGA 2011	15 МБ	95 Вт

Для сокетa LGA 1156 можна придбати материнську плату **GIGABYTE GA-H55M-S2**.

Характеристика материнської плати **GIGABYTE GA-H55M-S2**:

- Назва бренда: muchisimo
- Роз'єми: VGA
- Роз'єми: USB 2.0
- Роз'єми: PS2
- Роз'єми: Ethernet
- Роз'єми: Разъем PCI-Express X16
- Роз'єми: Разъем PCI-Express X1
- Роз'єми: SATA
- Чіпсет: Intel H55
- Тип Роз'єму: Разъем 1156
- Тип пам'яті: DDR3
- Застосування: Настільний комп'ютер
- Інтерфейс жорсткого диска: SATA
- Канал пам'яті: Подвійний
- Тип процесора: Intel
- Номер моделі: GA-H55M-S2
- Тип відеокарти: Дискретна
- Максимальний об'єм оперативної пам'яті: 8 ГБ
- Виробник чіпсета: Intel
- Форм-фактор: Микро-ATX
- Запам'ятовуючий пристрій: 2 DDR3 DIMM
- Product Brand: Gigabyte
- Ціна: \$37.07

Для сокетa LGA 771 (який підходить до материнських плат із сокетом LGA775) можна придбати материнську плату **Asus P5KPL-AM SE**

Характеристика материнської плати **Asus P5KPL-AM SE**:

- Назва бренду: Asus
- Роз'єми: VGA
- Роз'єми: USB 2.0
- Роз'єми: PS2
- Роз'єми: Ethernet
- Роз'єми: Роз'єм PCI-Express X16
- Роз'єми: Роз'єм PCI-Express X1
- Роз'єми: SATA
- чіпсет: Intel G31
- Тип роз'єму: LGA 775
- Тип пам'яті: DDR2
- Застосування: Настільний комп'ютер
- Інтерфейс жорсткого диска: SATAII
- Канал пам'яті: Подвійний
- Тип процесора: Core 2 Duo / Pentium / Celeron D / Pentium D
- Номер моделі: P5KPL-AM SE
- Тип відеокарти: Вбудований
- Максимальний об'єм оперативної пам'яті: 4 ГБ
- Виробник чіпсет: Intel
- Форм-фактор: U ATX
- USB-: 2 DDR2 DIMM
- Ціна: \$18.42

Для сокетa LGA 2011 можна придбати материнську плату **AtermiteX X58**.

Характеристика материнської плати **AtermiteX X58**:

- Назва бренду: Atermite
- Роз'єми: USB 2.0
- Роз'єми: PS2
- Роз'єми: Роз'єм PCI-Express X16
- Роз'єми: SATA
- чипсет: Intel X58
- Тип роз'єму: LGA 1366
- Тип пам'яті: DDR3
- Застосування: Сервер
- Інтерфейс жорсткого диска: 4 * SATA2.0
- Канал пам'яті: Одна
- Тип процесора: Intel
- Номер моделі: X58
- Максимальний об'єм оперативної пам'яті: 32 ГБ
- Виробник чипсет: Intel
- Форм-фактор: MATX
- USB-: 2 DDR3 DIMM
- Ціна: \$41.01

Вибір оперативної пам'яті дуже легкий, так як нам не потрібні складні розрахункові системи , достатньо буде 4 Гб оперативної пам'яті DDR2. Наприклад оперативна пам'ять від фірми **Kingston**, придбати дві планки по 2 Гб, щоб зробити роботу оперативної пам'яті в двох каналах.

Характеристика однієї планки оперативної пам'яті:

Об'єм (Гб): 2GB

Частота:800MHz

Ціна: \$4.99 * 2 = \$9.98

Стосовно відео карти можна обрати будь яку ,оскільки на неї не буде великого навантаження, вона потрібна тільки для запуску терміналу.

Як приклад звичайної відео карти по характеристикам , це 512 Мб пам'яті, без додаткового живлення.

PCI-E GeForce 9600 GT характеристики:

- графічний чіп GeForce: 9600 GT
- частота пам'яті: 1800 МГц
- Обсяг пам'яті: 512 МБ
- Розрядність шини пам'яті: 128 біт
- Тип пам'яті: GDDR3
- Максимально підтримуваний дозвіл: 2560x1600

Інші пристрої, характеристики котрих, не впливає на роботу системи:

- Монітор або термінал - ± \$14
- Мишка та клавіатура - ± \$8
- Корпус - ± \$4
- Охолодження процесора - ± \$7

1.8.3 Користувачі системи

Система розробляється під гуртожитки вищих навчальних закладів, технікумів, коледжів, домів типу гуртожиток, і т.п. то користувачами системи будуть це :

- Голова гуртожитку
- Комендант
- Директор студентського містечка

У кожного користувача системи буде своя роль, але доступ до системи в них однаковий. Один може допомагати іншому якщо якогось працівника немає.

1.8.4 Вибір операційної системи

На даному етапі, реалізація проекту буде виконана на операційній системі Windows XP, Windows 7, Windows 8, Windows 8.1, Windows 10.

Реалізація цього проекту на операційній системі windows 95, windows 98, windows vista, не є цілісообразним оскільки ці системи вже не підтримуються компанією Microsoft, і вже є занадто застарілими.

1.8.5 Вимоги до програмного забезпечення.

З метою зменшення витрати часу потрібно робити звіти в електронному вигляді, та мати можливість передавати їх в розширені яке найбільш поширене в суспільстві. Програмним забезпеченням Microsoft office Word, Microsoft office Exel.

Для збереження накопичуваних даних використовується програма Microsoft SQL server 2018. Реалізація графічного вигляду БД створене в середовищі Visual Studio 2019.

Реляційна база даних - це набір даних з зумовленими зв'язками між ними. Ці дані організовані у вигляді набору таблиць, що складаються із стовпців і рядків. У таблицях зберігається інформація про об'єкти, представлених в базі даних. У кожному стовпчику таблиці зберігається певний тип даних, в кожному осередку - значення атрибута. Кожна стока таблиці являє собою набір пов'язаних значень, що відносяться до одного об'єкту або сутності. Кожен рядок в таблиці може бути позначена унікальним ідентифікатором, званим первинним ключем, а рядки з декількох таблиць можуть бути пов'язані з допомогою зовнішніх ключів. До цих даних можна отримати доступ багатьма способами, і при цьому реорганізовувати таблиці БД не потрібно.

На думку авторів, яку базу даних робити, був очевидним. Зараз майже всі бази даних які створюються вони є реляційними. На 2019 рік , зі 100% баз даних які були створені лише 7% не реляційні.

Інтерфейс користувача

Інтерфейс користувача це половина проекту. Інтерфейс для користувача буде не складним. Складатися із декількох вкладок, та форм заповнення. Формування звітів винесено окремою вкладкою.

Збереження даних

Для захисту даних скористаємось паролем. Збереженні данні будуть доступні тільки користувачам, які були наведенні на початку цього пункту п.1.8.1.

Надійність

Вся інформація зберігатиметься на локальному сервері. Для кращого обліку та збереження логів потрібно поставити два жорстких диска. Зробити оновлення (копіювання на інший жорсткий диск) кожні 15-30 хв. Майже не можливо зробити так щоб два жорстких диска вийшли з ладу одночасно. А від стрибків напруги зберігає блок живлення самого комп'ютера або сервера.

1.8.6 Функції системи

З погляду на пункт .1.4. проект спрямований зменшити затрати часу при поселенні, і ввести облік якого до цього взагалі не було.

В проекті на контролях пропуску будуть використані картки студентів які служать пропуском в університеті. Таким чином не потрібно буде окремо здобувати пропуск.

До функцій які будуть реалізовані в системи можна віднести такі:

- Формування електронних даних студентів для поселення
- Формування списків розселення
- Ведення обліку проходження студентів через контроль пропуску
- Збереження інформації про облік студентів на SD карті

1.8.7 Вхідні , та вихідні данні

Для забезпечення прискорення процесу поселення вихідними даними мають бути готові документи які додаються до інших, тому потрібно розписати вхідні данні щоб розуміти, що ми отримаємо в кінці.

Документи передаються в гуртожиток з університету(деканату або приймальної комісії)

Документи студента (паспорт, ідентифікаційний, номер телефонну, адрес , тощо)

Вихідними даними з процесу поселення будуть , сформовані угоди, та звіти які додаються до наказу про поселення.

Вихідними даними обліку студентів , це збережені данні при проходженні крізь контролі пропуску, прізвище ім'я , по батькові , дата та час . А вхідними даними обліку, це картка студента з унікальним ключем.

РОЗДІЛ 2. Створення інформаційно-облікової системи поселення студентів у гуртожиток НУХТ

2.1 Розробка макета системи

Функціональний модель управління проектом «AS-IS»

Дана модель була розроблена за допомогою програми – AllFusion Process Modeler

Отже, на контекстній діаграмі А-0 (див. Додаток А рис.А.1) відобразимо основні вхідні потоки, вихідні дані, механізми та управління. Керуючими впливами тут будуть нормативно правова документація, накази і розпорядження, статут гуртожитку.

Функціональна модель складається з:

Вхідні дані:

- Житловий фонд ;
- Документи студентів;
- Наказ про зарахування;

Вихідні данні:

- База розселення;
- Наказ про поселення;
- Сформована угода;

Основними процесами для організації роботи поселення (див. Додаток А. рис. А.2.):

- аналіз списків зарахованих студентів що потребують поселення в гуртожиток;
- Розподіл кімнат між студентами;
- Формування ордерів;

- Прив'язка ордерів до пропуску;
- Формування наказу;

Процес розподіл кімнат між студентами декомпозиується (див. Додаток А.):

Розподіл кімнат між студентами:

- Облік вільних кімнат;
- Групування студентів за спеціальністю та курсом;
- Розселення по кімнатах;
- Пошук вільних місць;

Підселення студентів на вільні місця;

Модель TO-BE відрізняється від моделі AS-IS введеною інформаційною системою. За рахунок введення інформаційної системи як механізм, час на формування звітів та виконання/створення поселення знизилась, а в результаті підвищилась ефективність поселення.

Завдяки інформаційній системі, директор студентського містечка може контролювати облік оплати студентів, облік зайнятих або вільних кімнат.

Проект не є сильно важким для впровадження. Починаючи зі збору інформації до впровадження.

Спочатку потрібно ознайомитись з областю в якій потрібно робити цей проект. Після гарного ознайомлення с предметною областю, можна створювати модель AS-IS. Після додавання до недосконалої моделі AS-IS, модулів, функції ми доводимо цю модель до завершення, моделі TO-BE.

Коли вже створена модель TO-BE, переходимо до самої розробки проекту, а саме:

- Проектування бази даних.
- Створення інтерфейсу користувача.

Рис 2.1: Схема поселення2DataSet у вигляді XSD в Visual Studio 2019

Після під'єднання бази до середовища Microsoft Visual Studio 2019 можна починати роботу із розробкою форм для заповнення бази даних. Чим менше різних об'єктів ми створюємо тим зрозумілішим буде інтерфейс для користувача, але не потрібно його робити настільки скудним, що половину функцій які він має виконувати, будуть відсутні.

Інтерфейс користувача

Інтерфейс користувача - сукупність засобів для обробки та відображення інформації, максимально пристосованих для зручності користувача; у графічних системах інтерфейс користувача реалізовується багатовіконним режимом, змінами кольору, розміру, видимості (прозорість, напівпрозорість, невидимість) вікон, їхнім розташуванням, сортуванням елементів вікон, гнучкими налаштуваннями як самих вікон, так і окремих їхніх елементів (файли, папки, ярлики, шрифти тощо), доступністю багатокористувацьких налаштувань.

Інтерфейс має бути зрозумілим, і не загрожує інформацією яка на даному етапі не є важливою.

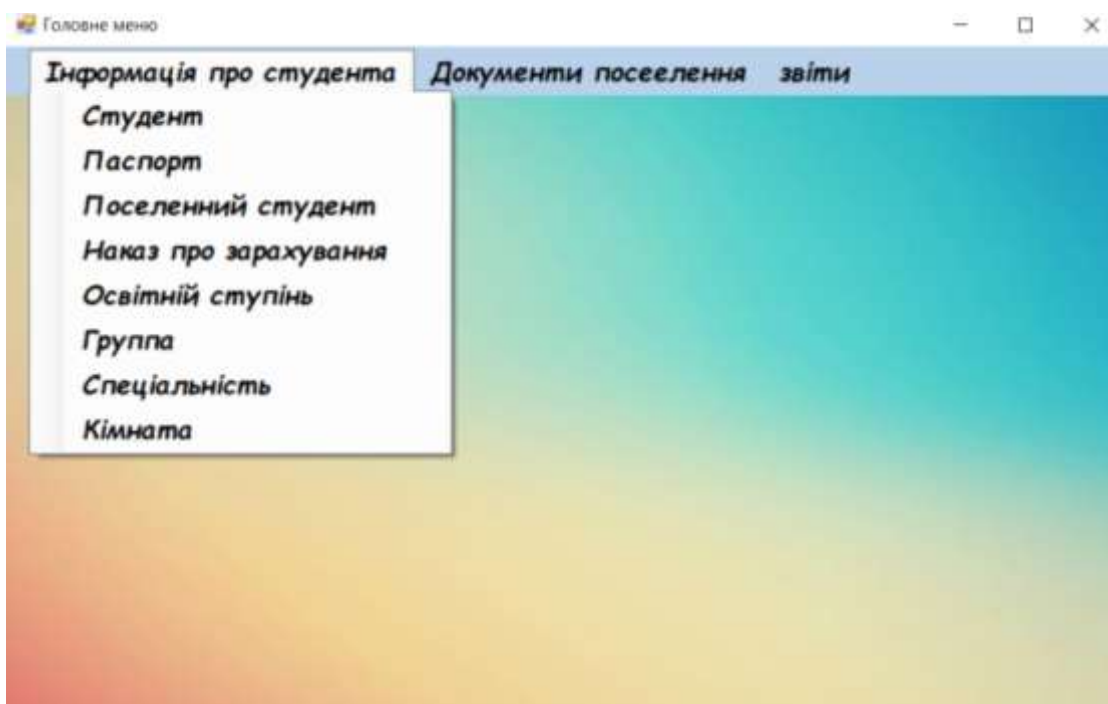


Рис.2.2 форма головного меню

На рисунку 2.2 ми бачимо форму «Головне меню». Інструмент `toolStripMenuItem` допомагає переходити до інших форм для заповнення, та додавання студентів до бази даних.

Кожна форма не є перегруженою тому вона буде зрозуміла для будь якої людини. Інші форми можна подивитися в додку «В»

На кожній формі є поля для реєстрації які складаються з елементів `Label`, елементу `ComboBox` та елементу `TextBox`.

На формі з дочірніми таблицями додаємо `dataGridView`

Створюємо таких форм приблизно по кількості існуючих таблиць , і ще декілька форм для формування звітів.

Інструкція користувача

Ця система розробляється для прискорення процесу поселення в гуртожиток студентів НУХТ. Користувачі системи є голова гуртожитку, комендант, директор студентського містечка. Дивитися пункт 1.8.1

Інсталювати програму не потрібно. Запустивши файл з розширенням file_name.exe запускається Головне меню. Користувач може обрати пункт меню та при натисканні на цей пункт йому відкриється нова форма, для додавання, збереження, видалення інформації.

Кожний пункт меню відповідає за окрему форму на якій розміщено форму для заповнення окремої таблиці.

Форм для заповнення інформації про студента загалом вісім.

На головній формі для користувача зверху в елементі menuStrip є **три** розділи в кожному з яких є декілька підпунктів.

Перший розділ має вісім підпунктів, кожен з яких відповідає за окрему таблицю для заповнення.

Відсортовані ці пункти по мірі їх потреби (більш часто або менш часто) ось в такому порядку:

- Студент
- Паспорт
- Поселений студент
- Наказ про зарахування
- Освітній ступінь
- Група
- Спеціальність
- Кімната

При переході на форму «Студент» у новому вікні відкривається ця форма. Складається з двох дочірніх dataGridView таблиць для перегляду зареєстрованих записів та однієї таблиці для заповнення, збереження або видалення записів. Складається з декількох елементів Label, ComboBox та TextBox. Усі форми які надалі будуть описані, винесені у додаток «В».

Розділ «Паспорт» це форма для заповнення паспортних даних студента. Форма складається з одного dataGridView та основної таблиці яка складається з декількох елементів Label, ComboBox та TextBox.

В розділі «Поселений студент» як і в більшості інших майже все таке саме. Форма складається з двох dataGridView. Вони потрібні для перегляду записів. Таблиця заповнення складається з декількох елементів Label, ComboBox, TextBox та DateTimePicker, останній інструмент потрібен для виставлення дати поселення та виселення.

Розділи «Наказ про зарахування» та розділ «Група» має такий самий принцип як і попередні розділи. Різниця в кількості dataGridView таблиць. Для збереження заповнення створено по одній таблиці в обох формах.

На головній формі є розділи які мають тільки таблицю для заповнення. Розділи «Освітній ступінь», «Спеціальність», «Кімната». Ці всі форми є найменш використовувані тому вони не мають дочірніх таблиць.

Другим елементом в інструменті menuStrip є розділ «Документи поселення». Розділи «Угода» «Пошук». При натисканні на «Угода», відкриється форма для пошуку вже зареєстрованих студентів. Пошук відбувається за двома параметрами, це дата вступу до вищого навчального закладу, або за номером наказу на вступ. Розділ «пошук» виконує пошук студента по прізвищу.

Останній елемент на інструменті menuStrip це розділ «Звіти».

Цей розділ як і попередній має два підрозділи. Один це Готова угода. Ще одним підрозділом є «Звіт».

На формі «Готова угода» декілька полів Label, ComboBox, TextBox, які потрібні для створення швидкої сформованої угоди. Вигляд форми сформована угода можна подивитися у додатку «С».

2.3 Вибір елементної бази

Враховуючи вимоги електронної системи необхідно підібрати комплектуючі. Для реалізації апаратної частини облікової системи проходження крізь контроль пропуску необхідні наступні комплектуючі:

- ✓ Мікроконтролер
- ✓ Сканер
- ✓ Блок живлення
 - Акумулятор
 - Контролер заряду АКБ
- ✓ Корпус
- ✓ Роз'єми

2.3.1 Вибір мікроконтролера

На даний час на ринку існує чимала кількість мікроконтролерів різних виробників. Найпопулярніші представлені на рис. 2.1 – рис.2.4.



Рис. 2.3.1 Arduino Mega



Рис. 2.3.2 Arduino Uno

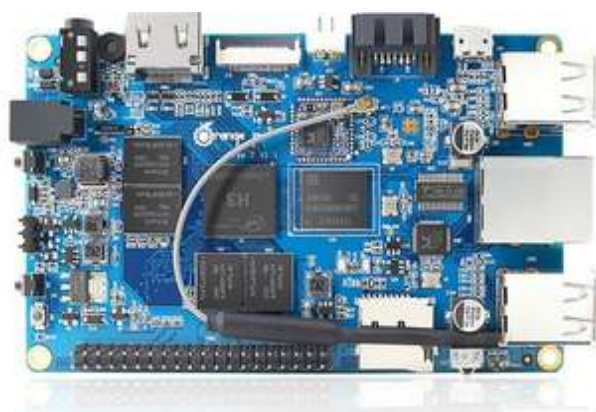


Рис. 2.3.3 Orange pi



Рис. 2.3.4 Raspberry pi

На рис. 2.5 зображено мікроконтролер ESP8266 з модулем Wi-Fi, що підтримує стандарт 802 b,g,n.

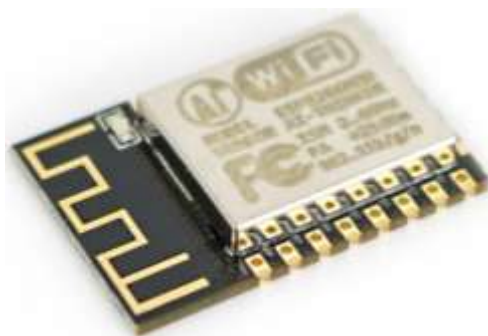


Рис. 2.3.5 Модуль ESP8266

Основними критеріями вибору мікроконтролеру є:

- ✓ Висока відмовостійкість
- ✓ Доступність комплектуючих
- ✓ Низька напруга живлення (до 5в)
- ✓ Мінімальне споживання електроенергії модулем
- ✓ Наявність Wi-Fi модуля
- ✓ Компактність
- ✓ Низька вартість комплекту

Необхідні характеристики мікроконтролерів представлені в таблиці 2.1

Таблиця 2.1 порівняння мікроконтролерів

Мікроконтролер	Наявність Wi-Fi	Напруга живлення	Вартість грн	Компактність
Arduino Mega2560	-	5V+	310	-
Arduino Uno	-	5V+	125	-
Arduino Nano	-	3.3V+	89	+
Orange pi	+	5V+	477	-
RASPBERRY PI	+	5V+	1500	-
ESP8266	+	3.3V+	85	+

Отже, враховуючи вимоги найкраще підходить модуль ESP 8266

На рисунку 2.6 зображено платформу NodeMCU.

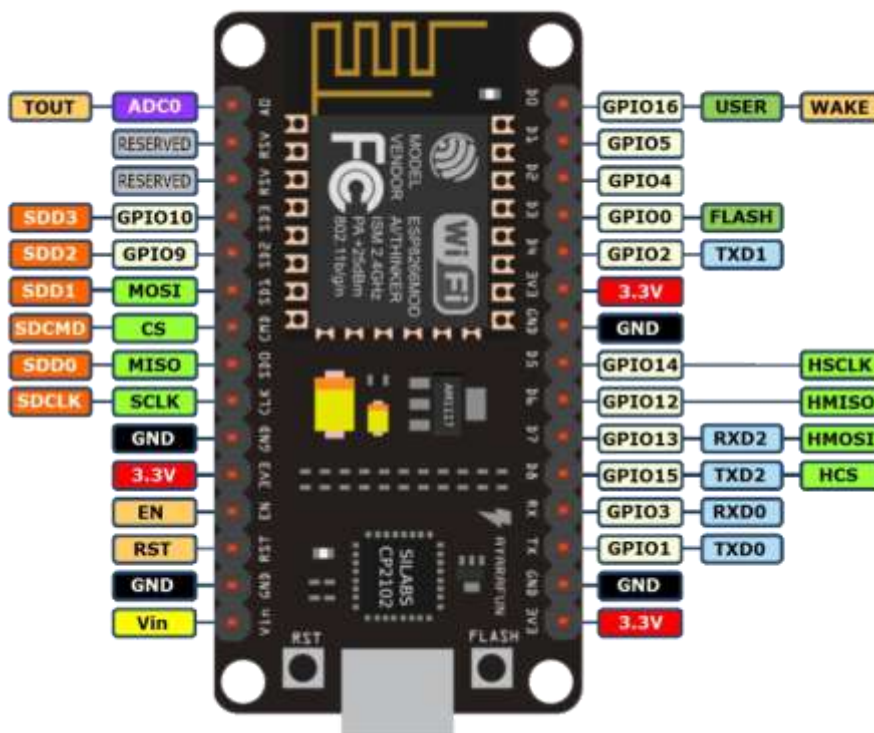
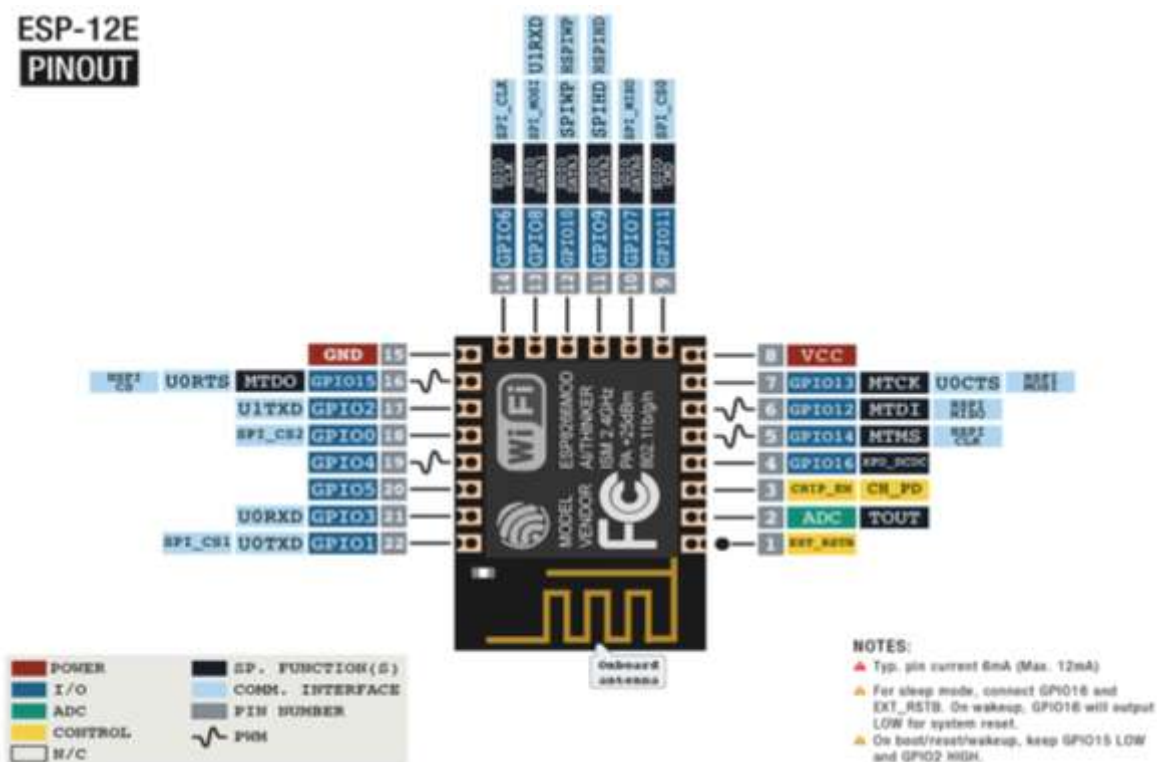


Рис.2.3.6 Плата nodemcu



ESP8266 - мікроконтролер китайського виробника Espressif з інтерфейсом Wi-Fi . Крім Wi-Fi мікроконтролер відрізняється можливістю виконувати програми з зовнішньої флеш-пам'яті з інтерфейсом SPI .

Модуль виконаний у форматі **ESP-12F** і є просунутою версією модуля ESP8266-12 з узгодженим вихідним каскадом і антеною. Особливістю модуля є більш економічна робота в активному режимі і ультра низьке споживання в сплячому і режимі очікування. На модулі виведені призначені для користувача виходи GPIO, що ще більше розширює можливості побудови на даному модулі систем "розумний будинок" і "розумних речей".

Даний модуль має антену виконану на друкованій платі і підписані виходи, що полегшують його використання. На відміну від молодших моделей має більшу кількість портів GPIO, а також режим сну з низьким споживанням живлення.

Характеристики:

- Модуль: ESP8266MOD
- Сумісність: ESP-12
- Обсяг флеш-пам'яті: 4Мб
- Оперативна пам'ять: 80Кб
- Напруга живлення: 3.3В
- Струм в режимах:
 - Глибокого сну: <10uA
 - Повної зупинки: <5uA
- Робоча частота: 2.4 ГГц
- Потужність випромінювання: + 24dbm
- Порти GPIO: 12
- Тип антени: вбудована, на друкованій платі
- Бездротові режими: точка доступу / softAP / SoftAP + точка доступу;
- Бездротовий протокол: 802.11 b / g / n;

- Підтримка безпеки: Wi-Fi @ 2.4GHz, підтримка WPA / WPA2 режиму безпеки
- Управління: вбудованими AT-командами

Існує два варіанти оновлення прошивки мікроконтролера: підключивши безпосередньо до комп'ютера та оновлення через бездротову мережу Wi-Fi.

ОТА якщо бути точнішим FOTA - Firmware Over The Air в перекладі з англійської «прошивки по повітрю» - це офіційне оновлення (прошивки) пристрою за допомогою мережі інтернет.

Це процес завантаження вбудованого програмного забезпечення до модуля ESP за допомогою з'єднання Wi-Fi, а не послідовного порту. Така функціональність стала надзвичайно корисною у разі обмеженого або відсутності фізичного доступу до модуля.

В таблиці 2.2 відображено за якою адресою в пам'яті знаходяться компоненти прошивки.

Таблиця 2.2 розподіл пам'яті в мікроконтролері ESP8266

ESP8266			
SMI Flash ROM Layout (без оновлення OTA)			
Адреса	Розмір	Ім'я	Опис
00000h	248k	app.v6.flash.bin	Додаток користувача
3E000h	8k	master_device_key.bin	ОТА. Не підтверджено: Не використовується без OTA
40000h	240k	app.v6.irom0text.bin	Бібліотеки SDK
7C000h	8k	esp_init_data_default.bin	Налаштування по замовчуванню
7E000h	8k	blank.bin	Заповнений FFh. Може бути конфігурація WiFi
SPI Flash ROM Layout (з оновленням OTA)			
Адрес	Размер	Ім'я	Опис
00000h	4k	boot.bin	Завантажувач
01000h	64k	app.v6.flash1.bin	Додаток користувача, слот 1
11000h	180k	app.v6.irom0text1.bin	Бібліотеки SDK
3E000h	8k	master_device_key.bin	Ключ пристрою OTA
40000h	4k		Не використовується
41000h	64k	app.v6.flash1.bin	Додаток користувача, слот 2
51000h	180k	app.v6.irom0text1.bin	Бібліотеки SDK, слот 2
7E000h	8k	blank.bin	Заповнений FFh. Може бути конфігурація WiFi

2.3.3 Вибір інших комплектуючих

Акумулятор обиралися відповідно вимогам живлення мікроконтролера. Враховуючи відношення розмір – ємність був обраний li-ion акумулятор зображений на рис.2.8.



Рис. 2.3.8 Акумулятор li-ion 18650 samsung

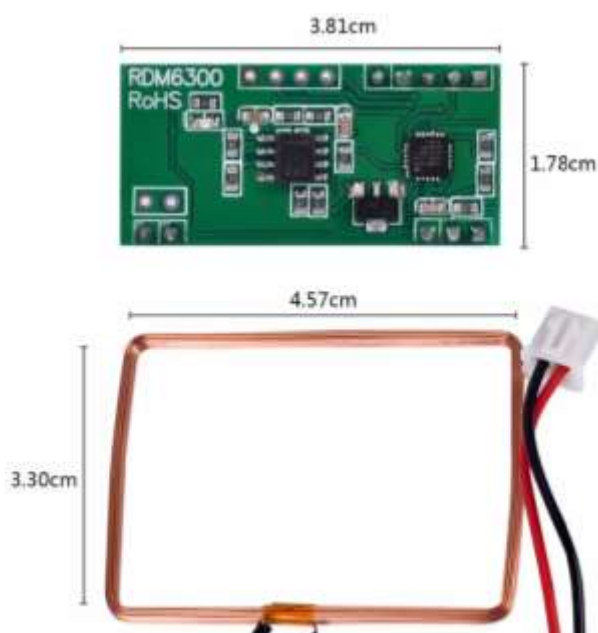


Рис. 2.3.9 сканер для считування карток rdm6300

Характеристики сканера та опис RDM6300

RDM6300 - безконтактний зчитувач призначений для дистанційного зчитування номера (ID) RFID брелка або карти стандарту «EM4100», що працюють на частоті 125 кГц і передачі цього номера через послідовний інтерфейс UART (TTL) на контролер, який керує електрозамком в системах контролю доступу в будинках, квартирах, гаражах і інших об'єктах.

«RDM6300» можна встановлювати приховано в стіну або в корпус для встановлення на стінах. В якості контролера системи доступу можна використовувати мікроконтролер «Arduino», що відкриває електромагнітний або електромеханічний замок за допомогою силового ключа на польовому транзисторі.

Характеристики:

- Робоча частота: 125 кГц
- Бітрейт: 9600 (TTL рівень RS232)
- Інтерфейс: Wiegand26 або TTL рівень RS232
- Струм: <50 мА
- Діяльність зчитування: 20 ... 50 мм (в залежності від карти / брелка, виробника)
- Напруга живлення постійного струму: 5 В
- Діапазон робочих температур: -10 ... + 70 ° С
- Розміри плати: 38.5 x 19 x 9 мм
- Розміри антени: 45 x 35 x 3 мм

Підключення «RDM6300»:

Роз'єм P1:

- «TX» - передача даних
- «RX» - прийом даних

не застосовується

- «GND» - загальний провід
- «+ 5V» - «+5 В» постійного струму

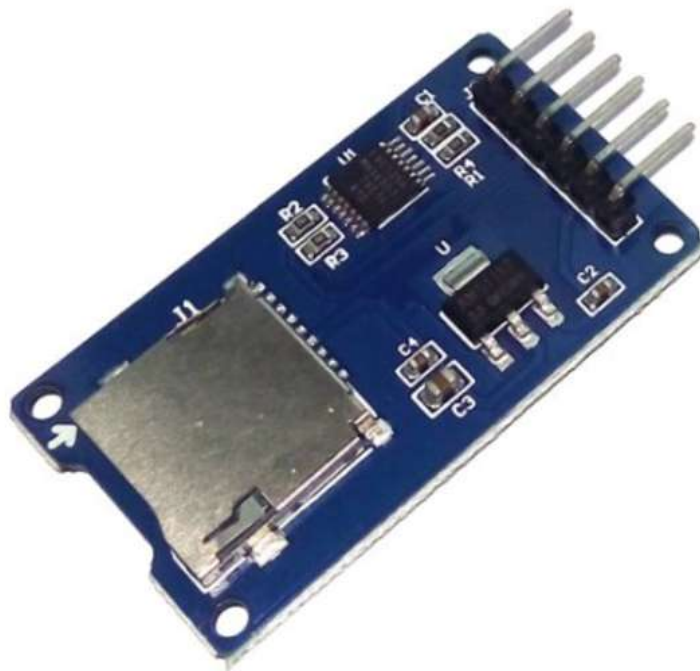
Роз'єм P2:

- ANT1 - підключення антени
- ANT2 - підключення антени

Роз'єм P3:

- LED
- + 5V (DC) - «+5» В постійного струму
- GND - загальний

Модуль micro SD карт, SPI



Характеристики та призначення

- Підтримка карт: micro SD, micro SDHC
- Інтерфейс: SPI
- Напруга живлення: 4.5-5.5 В або 3.3 В
- Напруга інтерфейсу SPI: 5 В або 3.3 В
- Струм: від 0.2 мА до 200 мА
- Підтримувані карти: Micro SD карти (≤ 2 Г), Micro SDHC карти (≤ 32 Г)
- Габарити: 42мм x 24мм x 12мм

Використання SD карти необхідно, оскільки у разі знеструмлення будівлі, система деякий час зможе працювати від універсальних мобільних батарей які були зображені на рисунку 2.8. та зберігати дані на SD карту.

Огляд середовища Arduino IDE для реалізації програмної частини

Arduino IDE

Програмне забезпечення Arduino із відкритим вихідним кодом (IDE) дозволяє легко писати код та завантажувати його в мікроконтроллер. Він працює на Windows, Mac OS X і Linux. Arduino IDE написана на Java.

Головне вікно програми Arduino IDE зображене на рисунку 2.10.

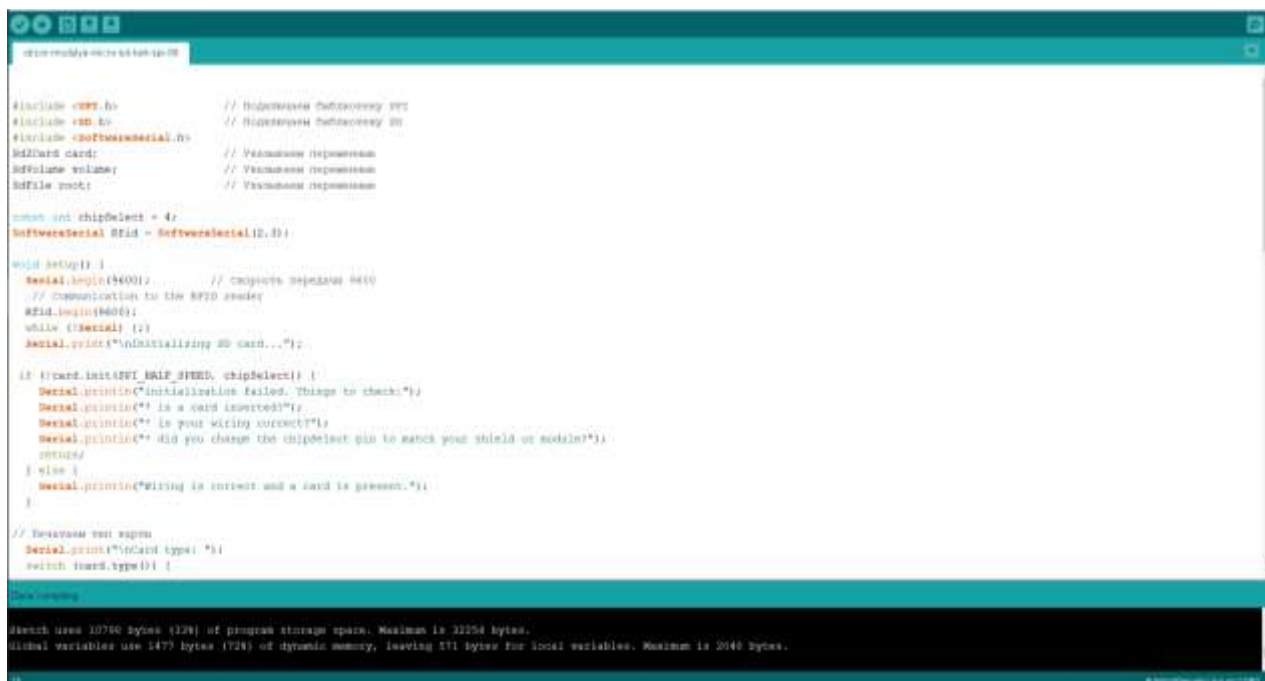


Рис. 2.3.10 Інтерфейс середовища розробки Arduino IDE

При натисканні на кнопку «verify» проходить компіляція скетчу та бачимо в консолі програми, яка знаходиться з низу, помилок не має.

Web інтерфейс

Дані які були відскановані модулем rdm6300 імпортуються на web-сервер. В онлайн додатку google sheets (таблиці). Вони мають автоматично заповнюватись.

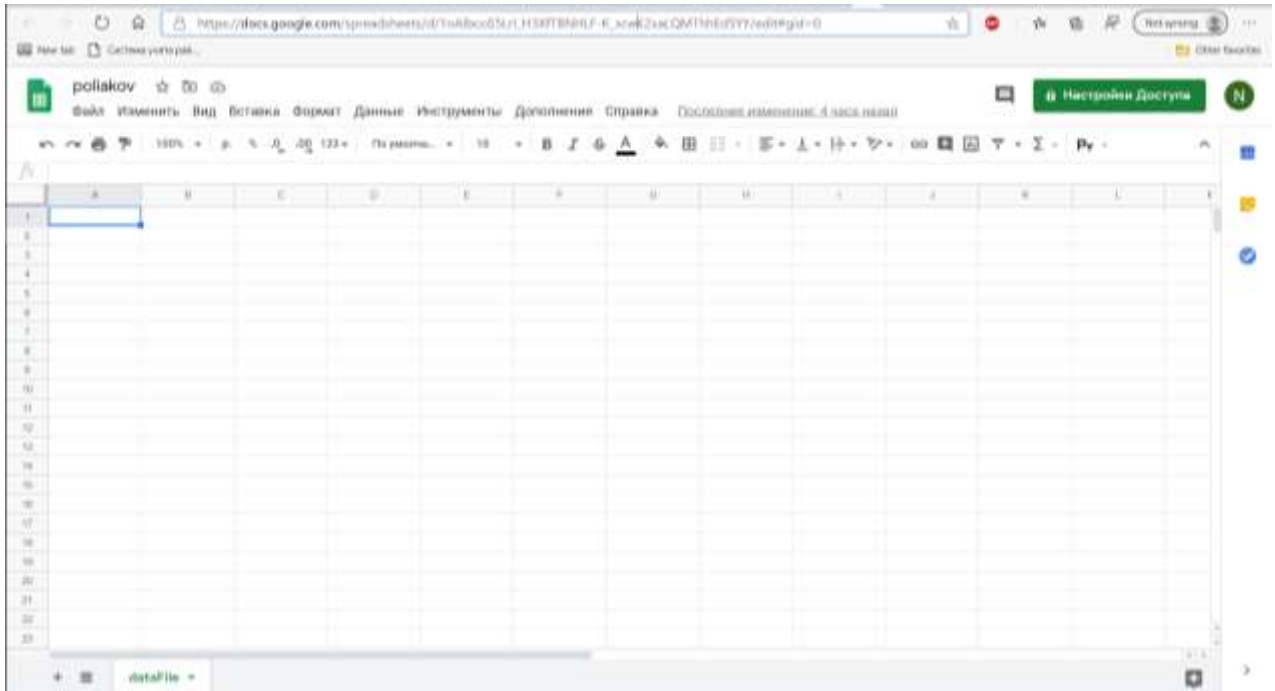


Рис.2.3.11. форма web-інтерфейсу

Крім коду для відправлення інформації з модулів на сайт, на формі сайту також є код скрипта обробки інформації. Щоб не нагрудать основний розділ проекту , код скрипта буде винесений у додаток «С».

Розділ 3 Охорона праці

НПАОП 80.0-1.12-04	Правила безпеки під час навчання в кабінетах інформатики навчальних закладів системи загальної середньої освіти
--------------------	---

НПАОП 92.5-1.03-97	Правила охорони праці для культурно-освітніх закладів, що розташовані в сільській місцевості
НПАОП 93.0-1.06-97	Правила охорони праці при експлуатації пралень і лазень
НПАОП 93.0-1.07-97	Правила охорони праці для перукарень

Система управління охороною праці

1. Загальні положення

1.1. Система управління охороною праці (далі – СУОП) розроблена відповідно до вимог ст.13 Закону України «Про охорону праці», ДБН А.3.2-2-2009 «Охорона праці і промислова безпека у будівництві» (НПАОП 45.2-7.02-12), ДСТУ OHSAS 18001:2010. «Системи управління гігієною та безпекою праці», ДСТУ OHSAS 18002:2015 «Системи управління гігієною та безпекою праці.

Основні принципи виконання вимог OHSAS 18001», ДСТУ ГОСТ 12.0.230:2008 «Система стандартів безпеки праці. Системи управління охороною праці. Загальні вимоги», «Рекомендацій щодо побудови, впровадження та удосконалення системи управління охороною праці» (затверджено Держгірпромнаглядом України 07.02.2008) та інших законодавчих і нормативно-правових актів з питань охорони праці, з метою створення на робочому місці умов праці відповідно до нормативно-правових актів, а також для забезпечення додержання вимог законодавства щодо прав працівників у галузі охорони праці.

1.2. СУОП - це сукупність взаємопов'язаних елементів, які відображають політику і цілі охорони праці та процедури досягнення цих цілей. Сутність СУОП полягає у створенні комплексної системи профілактики небезпечних

ситуацій, що виникають у процесі виконання будівельно-монтажних робіт, попередженні і мінімізації виробничих небезпек, ризиків, матеріальних збитків.

1.3. СУОП є підсистемою системи управління адміністративною і господарською діяльністю будівельної організації, що орієнтована на проведення попереджувальних дій, які запобігають виникненню небезпечних ситуацій, і дозволяє кожному рівню виробничого ризику протиставити матеріальний (технічний, фізичний, хімічний тощо) або нематеріальний (регламентація, профвідбір, навчання тощо) бар'єр, перешкоду.

Дія СУОП поширюється на всі робочі місця підприємства:

1.4. Загальне керівництво роботами із створення в будівельній організації СУОП і забезпечення її функціонування покладається на роботодавця, а організаційно-методична робота з управління охороною праці - на службу охорони праці.

5.6. Організаційна структура СУОП базується на системі управління і має містити такі основні елементи:

- об'єкт управління;
- інформацію про стан об'єкта управління;
- орган управління; - управлінські впливи;
- пам'ять системи;
- зовнішні впливи;
- обмеження;
- інформаційну та звітну документацію.

5.7. Робітники повинні бути ознайомлені з Положенням про СУОП, яка має бути колективом підприємства обговорена та ухвалена під час підписання колективного договору.

2. Загальний опис СУОП

2.1. СУОП призначена забезпечувати охорону життя, здоров'я та безпеку працівників, у тому числі тимчасових працівників, інших осіб на підприємстві.

2.2. Управління охороною праці є процес:

- збирання, передавання, опрацювання інформації про стан безпеки праці у гуртожитках та виробничих підрозділах;
- підготування і прийняття управлінських рішень стосовно ліквідації виявлених порушень вимог норм і правил охорони праці, про що повідомляється об'єкту управління.

2.3. Політика СУОП охоплює:

- планування заходів з охорони праці
- запровадження та функціонування СУОП з урахуванням розгалуженої організаційної структури;
- перевіряння;
- аналізування керівництвом;
- постійне вдосконалення;
- здійснення корегувальних та попереджувальних дій;
- можливість адаптації до обставин, що змінилися;
- можливість інтеграції в загальну систему управління

2.4. Управління охороною праці забезпечується виконанням комплексу таких взаємопов'язаних функцій:

- планування діяльності структурних і виробничих підрозділів, функціональних служб із забезпечення безпеки праці;

- організація виконання заходів із забезпечення безпеки праці і функціонування СУОП;
- контроль за станом охорони праці і функціонуванням СУОП;
- облік, аналіз, оцінка стану безпеки та охорони праці;
- стимулювання підвищення рівня безпеки праці.

Визначені функції реалізуються керівниками структурних і функціональних підрозділів спільно з фахівцями служби охорони праці і робітниками гуртожитку на принципах соціального партнерства.

2.5. Планування СУОП передбачає забезпечення:

- комплексних заходів, що забезпечують досягнення встановлених нормативів з охорони праці, підвищення існуючого рівня охорони праці;
- заходів з охорони праці, передбачених колективним договором;
- заходів щодо усунення недоліків і виконання приписів органів державного нагляду;
- заходів щодо поліпшення умов і безпеки праці, запропонованих профспілковими органами, комісіями з охорони праці та органами Держпраці;
- заходів щодо усунення недоліків, виявлених під час розслідування нещасних випадків та аварій;

2.6. Основні цілі функціонування СУОП досягаються:

- визначенням прав, обов'язків, зацікавленості та відповідальності всіх категорій працюючих стосовно дотримання норм і правил охорони праці;
- організацією ефективної системи навчання;

- безперервним і дієвим контролем за станом умов і безпеки праці на робочих місцях;
- запровадженням єдиної методики аналізу та оцінки ступеня безпеки, рівня ризику виробництва;
- стимулюванням зменшення травматизму і профзахворювань.

2.7. Критеріями ефективності функціонування СУОП є:

- поліпшення умов і безпеки праці;
- поліпшення технічних, санітарно-гігієнічних, психофізіологічних і санітарно-побутових умов праці;
- поліпшення профілактичної діяльності у сфері охорони і безпеки праці;
- зменшення кількості травмонебезпечних ситуацій і, як наслідок, зниження рівня травматизму і профзахворюваності.

2.8. Для забезпечення функціонування СУОП повинен бути розроблений комплекс документів, зокрема:

- функціональні та посадові обов'язки (від працівника до роботодавця);
- перелік порушень норм і правил безпеки праці;
- методика оцінки результатів діяльності у сфері безпеки та охорони праці;
- система стимулювання.

2.9. Запровадження та підтримання функціонування СУОП передбачає:

- визначення осіб, уповноважених на розроблення, запровадження та підтримання функціонування СУОП, а також на забезпечення її необхідними ресурсами;
- забезпечення необхідної компетентності (належної освіти,

підготовленості та достатнього досвіду роботи) кожного працівника, який виконує завдання, яке може спричинити вплив на безпеку праці;

- забезпечення інформаційного зв'язку з питань охорони праці на підприємстві, а також з підрядниками та іншими сторонніми особами;
- залучення працівників до робіт, виконуваних у межах СУОП, а також до обговорення важливих питань з охорони праці;
- розроблення та запровадження документації СУОП, підтримання документації в актуальному стані, а також забезпечення належного контролю за дотриманням вимог документів СУОП.

2.10. Перевіряння передбачає:

- дотримання закону України «Про охорону праці», нормативно-правових актів про охорону праці;
- стану умов праці працівників;
- виконання керівництвом своїх обов'язків щодо створення безпечних і здорових умов праці;
- забезпечення безпечного виконання робіт під час виконання посадових обов'язків.

2.11. Основні види контролю за охороною праці:

- контроль з боку керівника а також інженеру з охорони праці;
- цільові перевірки робочих місць;

Результати раптових перевірок робочих місць оформляються актом і щомісячно розглядаються керівником.

2.12. Аналізування з боку керівництва передбачає розгляд низки питань, що стосуються функціонування СУОП, з тим, щоб оцінити її придатність,

адекватність та результативність, для підготовки відповідних розпорядчих документів з питань функціонування та запровадження СУОП, а також визначення можливості поліпшення діяльності у сфері охорони праці.

2.12.1. Аналіз та оцінка стану охорони праці.

Для аналізу повинні використовуватися: матеріали розслідування нещасних випадків, професійних захворювань, аварій, приписи органів державного нагляду; матеріали спеціальних обстежень з питань охорони праці, дані паспортизації санітарно-технічного стану умов праці.

На підставі аналізу і оцінки стану охорони праці розробляються та реалізуються управлінські рішення і здійснюється планування роботи з охорони праці.

2.12.2. За успішне проведення роботи і виконання запланованих заходів з питань охорони праці, направлених на створення та підтримання безпечних умов праці на робочих місцях, з метою підвищення зацікавленості працівників передбачається їх стимулювання.

Стимулювання здійснюється на підставі об'єктивної оцінки стану охорони праці на робочих місцях.

Види стимулювання працівників за активну участь та ініціативу у здійсненні заходів щодо підвищення безпеки і за роботу без порушень правил безпеки визначаються колективним договором.

2.13. Постійне удосконалення СУОП.

2.13.1. Навчання, інструктажі та перевірка знань працівників з питань охорони праці проводяться відповідно до Положення про навчання і перевірку знань працівників з питань охорони праці.

2.13.2. Пропаганда безпечних методів праці проводиться з використанням усіх форм, зокрема:

- оформлення інформаційних стендів;
- вивчення та впровадження передового досвіду з охорони праці;
- проведення оглядів, семінарів.

2.13.3. Безпека виконання посадових обов'язків працівниками досягається шляхом приведення робочих місць у відповідність до вимог існуючих нормативно-правових актів з охорони праці.

2.13.4. Забезпечення безпечного стану експлуатації будівель і споруд, призначення відповідальних осіб та їх навчання в спеціалізованих навчальних закладах визначається відповідними розпорядчими документами.

2.13.5. Дотримання вимог санітарно-гігієнічних умов праці, зокрема усунення причин виникнення небезпечних та шкідливих факторів на робочих місцях із застосуванням ефективних засобів захисту; належним утриманням та експлуатацією машин, механізмів.

2.13.6. Забезпечення засобами індивідуального та колективного захисту повинно здійснюватися відповідно до діючих норм, установленого порядку їх видачі, зберігання, користування та згідно з трудовим договором.

2.13.7. Організація лікувально-профілактичного обслуговування працівників повинна передбачати попередні (при прийнятті на роботу) та періодичні медичні огляди працівників і проведення лікувально-профілактичних заходів щодо попередження захворювань.

2.13.8. Професійний відбір працівників за окремими спеціальностями передбачає встановлення фізичної та психофізіологічної придатності працівників до безпечного виконання робіт.

ВИСНОВОК

В бакалаврській роботі було розроблено інформаційно-облікову систему поселення студентів у гуртожиток НУХТ.

Розібравшись в аналогах було створено унікальний проект якого досі не було на ринку. В результаті глибокого системного аналізу аналогів що присутні на ринку України та світу дозволив створити унікальний авторський проект. Інформаційно-облікова система яка була частково реалізована на мікроконтролері ESP8266. Має можливість підключення до web-сервісу. Реалізована функція локального зберігання даних на SD карті. Web-інтерфейс та підключення до інтернету дозволять спостерігати за даними з будь якої точки світу.

Взявши за основу унікальну картку-ключ RFID по якій студент проходить в університет було розроблено авторську систему обліку. В системі було реалізовано наступні функції:

- Формування електронних даних для поселення студента.
- Формування звітів зареєстрованих студентів
- Ведення обліку проходження студентів крізь контролі пропуску якого до цього не було
- Збереження даних на SD карті в разі знеструмлення системи

Поєднавши всі етапи розробки системи, в кінці ми отримали робочий проект що зменшить затрати часу при поселенні, а саме автоматичне формування електронних документів для поселення студентів та підвищити безпечність проживання в гуртожитку.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- 1 Веб сторінка [Електрон. ресурс]// <https://bitbucket.org/> - Режим доступу: <https://bitbucket.org/Remnis/rfid-125khz/src/master/RfID-125.ino> - Назва з екрану.
- 2 Веб сторінка [Електрон. ресурс]// <https://ssbb.com.ua/> - Режим доступу: <https://ssbb.com.ua/uk/sistemy-kontrolya-dostupa/sistemi-obl%D1%96ku-robochogo-chasu/> - Назва з екрану.
- 3 Веб сторінка [Електрон. ресурс]// <https://funart.pro/> - Режим доступу: <https://funart.pro/12790-fony-dlja-prilozhenij-63-foto.html/> - Назва з екрану.
- 4 Веб сторінка [Електрон. ресурс]// <https://sites.google.com/> - Режим доступу: <https://sites.google.com/site/tz5103voinovakateryna/metodi-rozrobki-web-sajtiv> - Назва з екрану.
- 5 Веб сторінка [Електрон. ресурс]// <https://xn----7sbhgu4ahbanfnng.xn--p1ai/> - Режим доступу: <https://xn----7sbhgu4ahbanfnng.xn--p1ai/product/%D0%B1%D0%B5%D1%81%D0%BA%D0%BE%D0%BD%D1%82%D0%B0%D0%BA%D1%82%D0%BD%D1%8B%D0%B9-%D1%81%D1%87%D0%B8%D1%82%D1%8B%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C-rfid-%D0%BA%D0%B0%D1%80%D1%82-rdm6300-125/> - Назва з екрану.
- 6 Веб сторінка [Електрон. ресурс]// <https://ospanel.io/> - Режим доступу: <https://ospanel.io/docs/#ustanovka> - Назва з екрану.
- 7 Веб сторінка [Електрон. ресурс]// <https://stackoverflow.com/> - Режим доступу: <https://stackoverflow.com/ru/q/13050864> - Назва з екрану.
- 8 Веб сторінка [Електрон. ресурс]// <https://qna.habr.com/> - Режим доступу: <https://qna.habr.com/q/384345> - Назва з екрану.
- 9 Веб сторінка [Електрон. ресурс]// <https://circuitdigest.com/> - Режим доступу: <https://circuitdigest.com/microcontroller-projects/log-temperature-sensor-data-to-google-sheet-using-nodemcu-esp8266> - Назва з екрану.
- 10 Веб сторінка [Електрон. ресурс]// <http://digitrode.ru/> - Режим доступу: http://digitrode.ru/computing-devices/mcu_cpu/2335-otpravlyaem-dannye-s-

- datchika-v-google-sheets-s-pomoschyu-esp8266-nodemcu.html - Назва з екрану.
- 11 Веб сторінка [Електрон. ресурс]// <https://github.com/> - Режим доступу: <https://github.com/electronicsguy/ESP8266/tree/8edb50dcf6efc952ef4df590adec9f9a19b6f2e5/HTTPSRedirect> - Назва з екрану.
 - 12 Веб сторінка [Електрон. ресурс]// <https://arduinoplus.ru/> - Режим доступу: <https://arduinoplus.ru/gde-skachat-esp8266-lib-h/> - Назва з екрану.
 - 13 ArduinoIDE // [Електронний ресурс] // – Режим доступу: www.arduino.cc. - Назва з екрану.
 - 14 Marco Schwartz , Internet of Things with ESP8266, Paperback – July 29, 2016.
 - 15 Arduino, датчики и сети для связи устройств. 2-е издание - Том Иго
 - 16 Улли Соммер - Программирование микроконтроллерных плат ArduinoFreeduino – 2012.
 - 17 Wheat Dale. - Arduino Internals (Technology in Action) ,friendsoft apress– 2011.
 - 18 Robert Faludi - Building Wireless Sensor Networks with ZigBee, XBee, Arduino, and Processing – 2010.

ДОДАТОК «А»

Схема моделі TO-BE

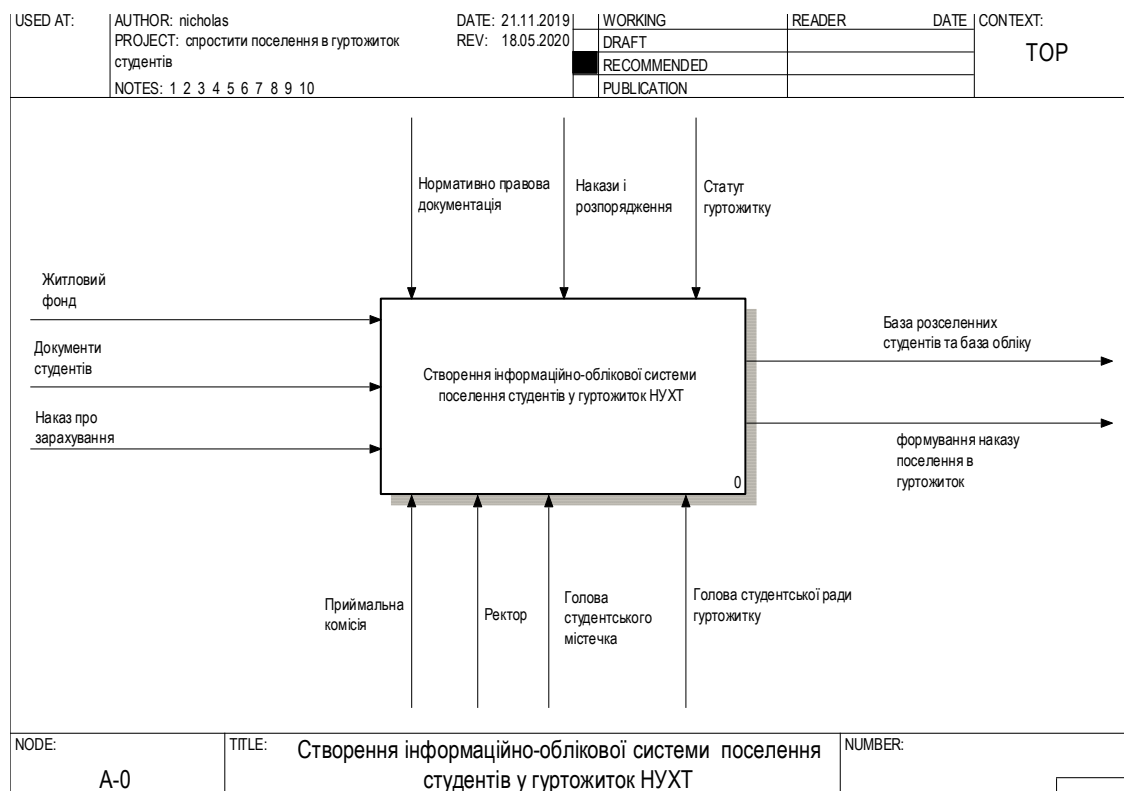


Рис.А.1 Модель TO-BE

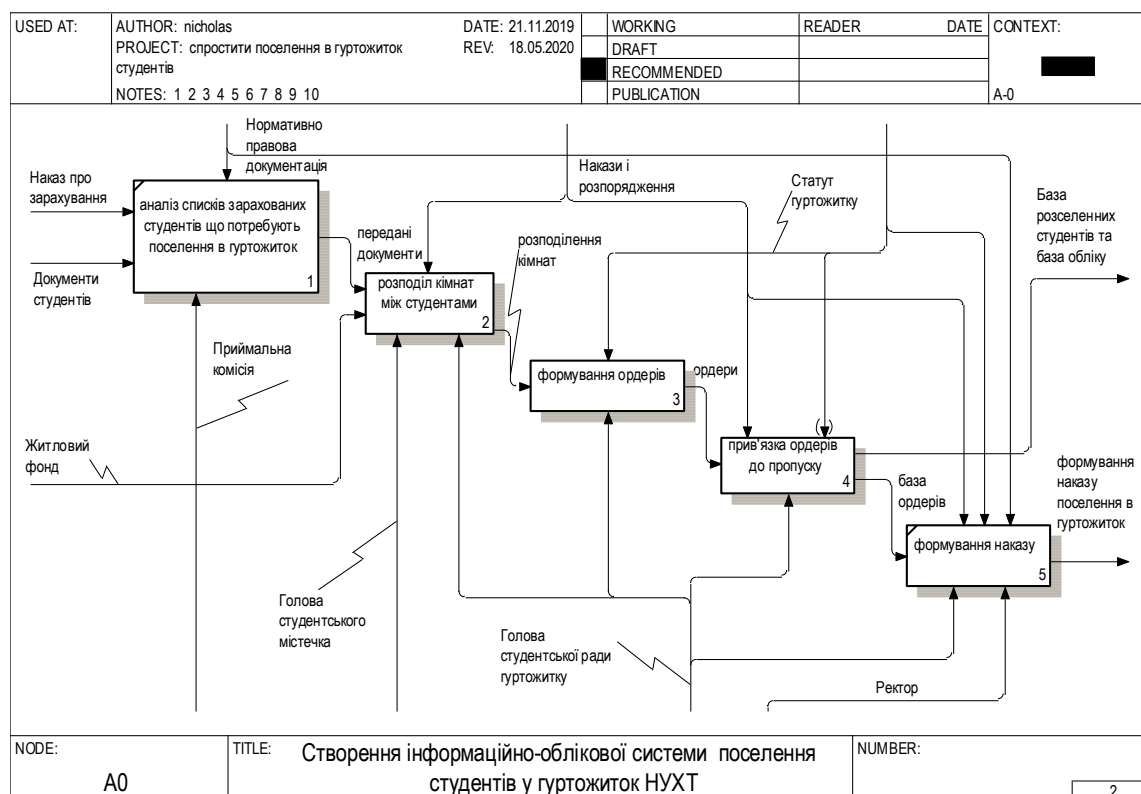


Рис.А.2. процеси для організації роботи поселення

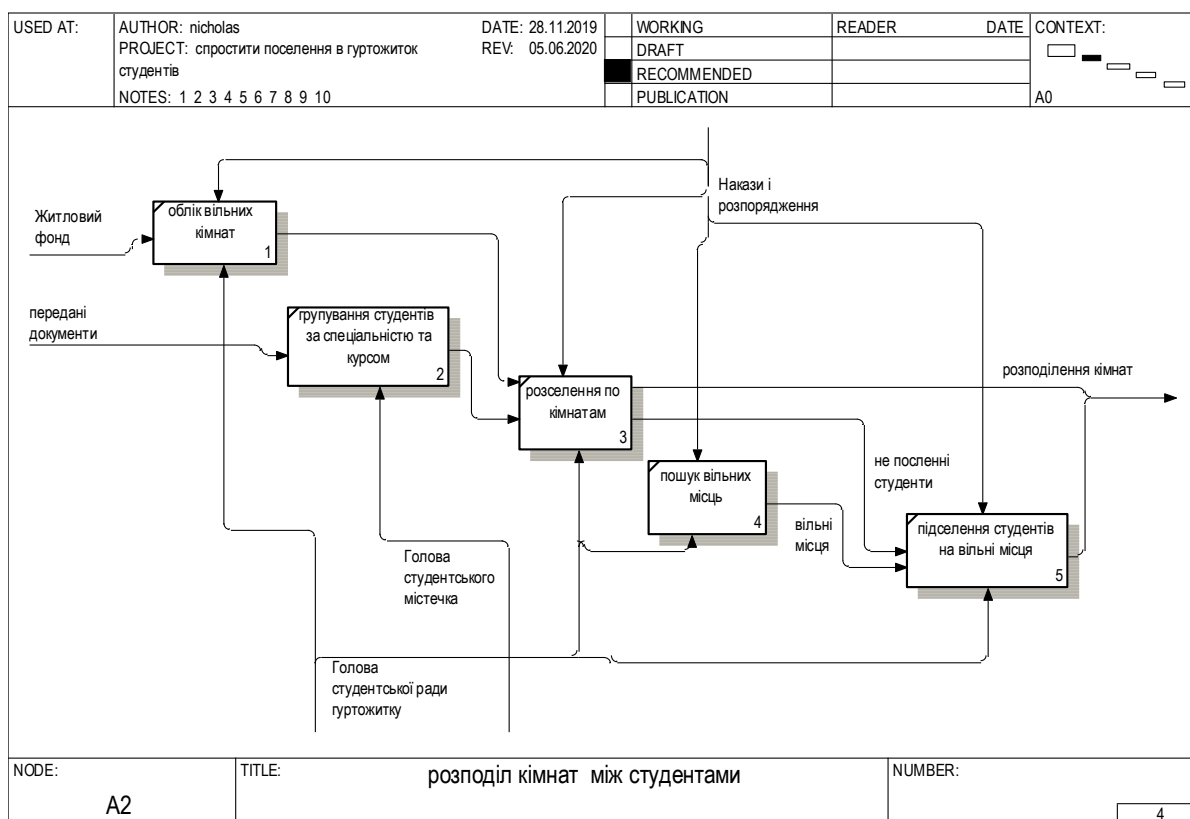


Рис.А.3. декомпозиція розподілу кімнат між студентами

ДОДАТОК «В»

кімната

1 of 4

Для додавання нової кімнати, її потрібно зареєструвати

Код кімнати: 1

Номер кімнати: 1

Поверх: 1

Кількість в кімнаті: 4

Рис.В.1. форма для реєстрації кімнати

наказ

1 of 6

Для того щоб створити наказ потрібно спочатку зареєструвати або обрати з існуючих спеціальностей та освітніх ступеней

Код наказу: 1

Номер наказу: 33

Дата наказу: 1 января 2015 г.

Кількість студентів: 10

Курс: 1

Код спеціальності: 1

Код ОС: 8

Код_ОС	Освітній_ступінь	Кількість
1	бакалавр	20
2	122	100
3	151	120
5	бакалавр	120
6	магістр	60
7	бакалавр	80
8	магістр	50

Код_спеціальності	Назва_спеціальності	Шифр
1	АК	151
3	ЮН	122

Рис.В.2 Форма для створення наказу

report

1 of 1 100% Find Next

Прізвище	Ім'я	Побатькові	Дата народження	Адрес	Номер телефона	Email	Ідентифікаційний	Код групи
Арсеній	Олена	Сергіївна	12/11/1999 12:00:00 AM	Полтава	0660436436	xmvelgg@sta.nl	3464262547	2
Бережний	Микола	Ігорович	9/14/1999 12:00:00 AM	Харків	0663253523	yzlusiio@supare.nl	5863835655	2
Загородній	Ярослав	Петрович	6/12/2000 12:00:00 AM	Слов'янськ	0660434847	wdftee@yomail.info	4288652725	2
Заріца	Віктор	Анатолійович	6/15/1999 12:00:00 AM	Дніпро	0952353256	chkeqci@netmail.tk	2575482458	2
Кілько	Петро	Петрович	5/21/1999 12:00:00 AM	Славянськ	0660434847	gnzpcic@netmail.tk	2353253252	2
Логанюк	Ігор	Петрович	12/12/1999 12:00:00 AM	Харків	0660434847	zbtwirc@netmail.tk	4326572572	1
Маланюк	Олена	Петрівна	2/13/1999 12:00:00 AM	Житомир	0660451221	elenas99@gmail.com	2352353523	2
Мамонтова	Ксенія	Олександрівна	5/14/1999 12:00:00 AM	Дніпро	0951004242	ksuadfb@emhub.com	4575475474	1
Міняєвський	Ярослав	Сергієвич	1/21/1998 12:00:00 AM	Слов'янськ	0660434847	ferqquitta@gmail.com	2198412848	3
Мудрий	Юрій	Юрійович	7/14/1999 12:00:00 AM	Біла Церква	0502363264	yeyugino@supere.nl	4573735475	2
Петренко	Петро	Данілович	6/14/2000 12:00:00 AM	Біла Церква	0660346436	menafhub@nipro.com	3475483434	1

Рис.В.3. Форма із звітом, як додаток до наказу

0 of 0

Код студента: 22

Прізвище: Поляков

Ім'я: Микола

Побатькові: Олександрович

Дата народження: 20 жовтня 1999 г.

Адрес: Слов'янськ

Номер телефона: 0503453454

Email: kolya99pp@gmail.com

Ідентифікаційний: 3811000123

Код наказу: 6

Код групи: 2

Рис.В.4 Форма для пошуку та корегування інформації

спец

1 of 2

Ви перейшли на форму для додавання та/або редагування спеціальності

Код спеціальності:

Назва спеціальності:

Шифр:

Рис.В.5 Форма для роботи зі спеціальностями

Форма Паспорт

Серія:

Номер:

Ким видано:

Дата видачі:

Термін дії:

Код студента:

№	Код студента	Прізвище	Ім'я	Поблискові	Дата народження	Адрес	Номер телефону	Email	Ідентифікаційна
22		Поліков	Микола	Олександр	20.06.1999	Славенск	0603453454	kolya1999@gmail.com	3011000123
23		Михайлович	Андрей	Сергійович	21.01.1998	Славенск	0660434847	sergii1998@gmail.com	2198412948
24		Маленкович	Олена	Петровна	13.02.1999	Житомир	0660451221	olena99@gmail.com	235253523
25		Клюк	Петро	Петрович	21.05.1999	Славенск	0660434847	petro21@netmail.it	2353253252
26		Тарасов	Павло	Сергійович	17.02.2000	Харків	0660434845	tarasov17@netmail.it	2353253252
27		Маленкова	Катерина	Олександрівна	14.05.1999	Дніпро	0951004242	katerina@netmail.it	4575475474
28		Петровський	Олександр	Олександрович	17.05.2000	Полтава	0660434847	oskars@superni.it	3454353457
29		Мудрий	Юрій	Юрійович	14.07.1999	Вінніца	0662363264	yuriy@superni.it	4573735475
30		Зарудка	Володимир	Анатолійович	15.06.1999	Дніпро	0952353256	chirp@netmail.it	2575483458
31		Загородній	Андрей	Петрович	12.06.2000	Славенск	0660434847	andrey@superni.it	4288852725
32		Березний	Микола	Іванович	14.03.1999	Харків	0663253523	yivko@superni.it	5063838355
33		Арсеній	Олена	Сергійівна	11.12.1999	Полтава	0660436436	oskars@superni.it	3454352547
34		Поліков	Ігор	Петрович	12.12.1999	Харків	0660434847	igor12@netmail.it	4326572572
35		Петренко	Петро	Дмитрович	14.06.2000	Вінніца	0660346436	oskars@superni.com	3475483434

Рис.В.6. Форма для заповнення паспорту

Код поселеного студента:

Дата поселення:

Код студента:

Код кімнати:

Код_студента	Номер_кімнати	Поворот
1	1	1
2	2	1
3	3	1
4	4	1

Код_студента	Прізвище	Ім'я	Поблисков	Дата_народження	Адрес	Номер_телефона	E-mail
21	Поліщук	Никола	Смоленський	23.08.1999	Смоленський	066343454	nikolap@ukr.net
23	Михайлюк	Олександр	Смоленський	21.01.1998	Смоленський	066343457	oleksandr@ukr.net
24	Михайлюк	Олена	Полтавський	13.02.1999	Полтавський	066343457	olena@ukr.net
25	Клиш	Петро	Полтавський	21.08.1999	Смоленський	066343457	petro@ukr.net
26	Тарасюк	Олена	Смоленський	17.02.2000	Київський	066343458	olena@ukr.net
27	Михайлюк	Олександр	Смоленський	14.08.1999	Дніпро	0957004242	oleksandr@ukr.net
28	Петренко	Олександр	Смоленський	17.08.2000	Полтавський	066343457	oleksandr@ukr.net
29	Михайлюк	Олександр	Смоленський	14.07.1999	Дніпро	066343458	oleksandr@ukr.net
30	Заручка	Віктор	Смоленський	16.08.1999	Дніпро	0957004242	viktor@ukr.net
31	Заручка	Олександр	Полтавський	12.08.2000	Смоленський	066343457	oleksandr@ukr.net
32	Заручка	Никола	Полтавський	14.08.1999	Київський	066343457	nikolap@ukr.net

Рис.В.7 Форма для студентів які поселяються в гуртожиток

ДОДАТОК «С»

Код Скрипта який використовується на Google Sheets

```

var SS = SpreadsheetApp.openById('1oAIbcoS5LrI_H3XfTBNHLf-
K_xcwK2xaLQMThhEd5YY'); //Enter Your Sheet ID Got From Sheet URL Link
var sheet = SS.getSheetByName('dataFile'); // Enter your sheet name here,
In my case it is TempSheet
var str = "";
function onOpen(){
    var ui = SpreadsheetApp.getUi();
    ui.createMenu('poliakov')
    .addItem('Clear', 'Clear')
    .addToUi();
}
function Clear(){
    sheet.deleteRows(4, sheet.getLastRow());
    SS.toast('Chart cleared', 'poliakov', 5);
}
function doPost(e) {
    var parsedData;
    var result = {};
    try {
        parsedData = JSON.parse(e.postData.contents);
    }
    catch(f){
        return ContentService.createTextOutput("Error in parsing request body: " +
f.message);
    }
    if (parsedData !== undefined){
        // Common items first
        // data format: 0 = display value(literal), 1 = object value
        var flag = parsedData.format;
        if (flag === undefined){
            flag = 0;
        }
        switch (parsedData.command) {
            case "appendRow":
                var tmp = SS.getSheetByName(parsedData.sheet_name);
                var nextFreeRow = tmp.getLastRow() + 1;
                var dataArr = parsedData.values.split(",");
                tmp.appendRow(dataArr);
                str = "Success";
                SpreadsheetApp.flush();
                break;
        }
        return ContentService.createTextOutput(str);
    } // endif (parsedData !== undefined)
    else{
        return ContentService.createTextOutput("Error! Request body empty or in
incorrect format.");
    }
}

```

```

    }
}
function doGet(e){
    var val = e.parameter.value;
    var cal = e.parameter.cal;
    var read = e.parameter.read;

    if (cal !== undefined){
        return ContentService.createTextOutput(GetEventsOneWeek());
    }
    if (read !== undefined){
        var now = Utilities.formatDate(new Date(), "EST", "yyyy-MM-dd'T'hh:mm
a'Z'").slice(11,19);
        //sheet.getRange('D1').setValue(now);
        //var count = (sheet.getRange('C1').getValue()) + 1;
        sheet.getRange('C1').setValue(count);
        return ContentService.createTextOutput(sheet.getRange('A1').getValue());
    }
    if (e.parameter.value === undefined)
        return ContentService.createTextOutput("No value passed as argument to
script Url.");
    var range = sheet.getRange('A1');
    var retval = range.setValue(val).getValue();
    var now = Utilities.formatDate(new Date(), "EST", "yyyy-MM-dd'T'hh:mm
a'Z'").slice(11,19);
    // sheet.getRange('B1').setValue(now);
    sheet.getRange('B1').setValue("Humidity");
    //sheet.getRange('C1').setValue('0');
    if (retval == e.parameter.value)
        return ContentService.createTextOutput("Successfully wrote: " +
e.parameter.value + "\ninto spreadsheet.");
    else
        return ContentService.createTextOutput("Unable to write into
spreadsheet.\nCheck authentication and make sure the cursor is not on cell
'A1'." + retval + ' ' + e.parameter.value);
}
function GetEventsOneWeek(){
    var Cal = CalendarApp.getCalendarsByName('Test REST API')[0];
    // Need to create 2 separate Date() objects. Cannot do 'OneWeekFromNow =
Nowjs' to
    // simply get it's value and use that later without modifying 'Now'
    // since in JS, an object is automatically passed by reference
    var Now = new Date();
    var OneWeekFromNow = new Date();
    OneWeekFromNow.setDate(Now.getDate() + 7);
    //Logger.log(Now);
    //Logger.log(OneWeekFromNow);
    var events = Cal.getEvents(Now, OneWeekFromNow);
    //Logger.log(events.length);
    var str = '\nEvent Title,\tDescription,\tRecurring?,\tAll-day?,\tFirst
Reminder (in minutes before event)\n';
    for (var i = 0; i < events.length; i++){

```

```
        str += events[i].getTitle() + ',\t' + events[i].getDescription() + ',\t' +
events[i].isRecurringEvent() + ',\t' + events[i].isAllDayEvent() + ',\t' +
events[i].getPopupReminders()[0];
        str += '\n';
    }
    //Logger.log(str);
    return str;
}
```

ДОДАТОК «D»

Додаток ERWin схем

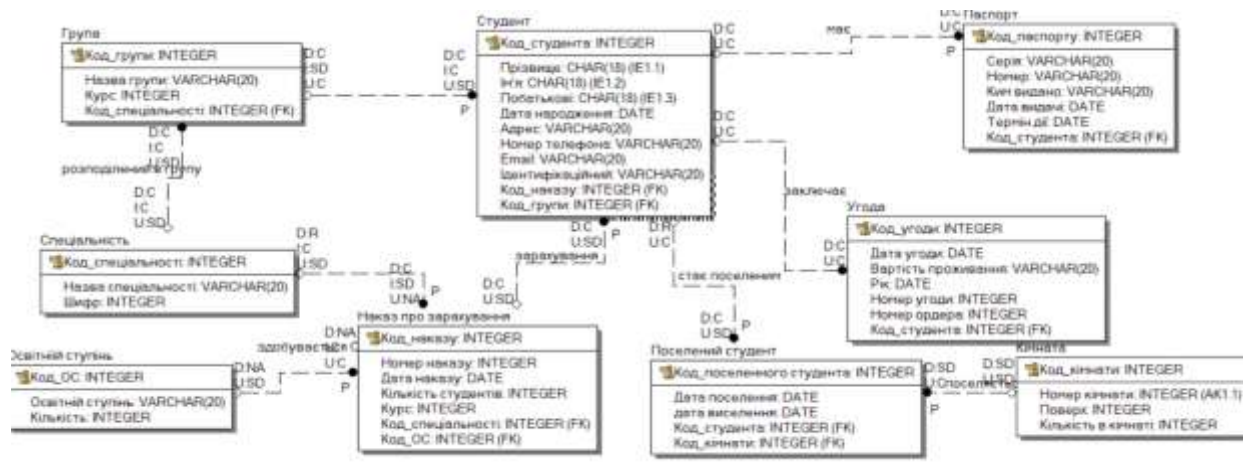


Рис.D.1. Логічна модель бази даних

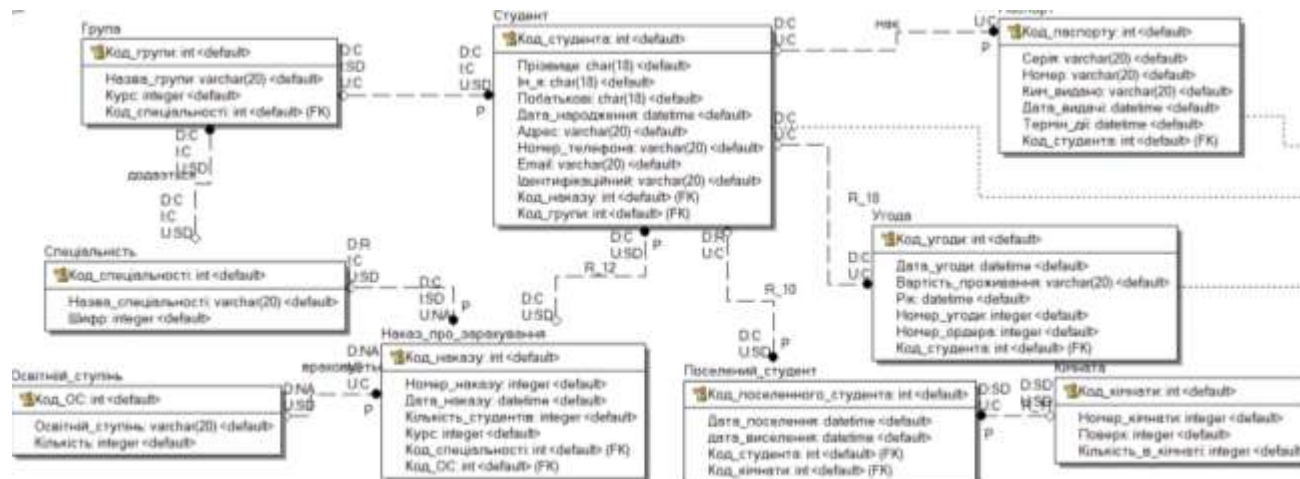


Рис.D.2. Фізична модель бази даних

ДОДАТОК «G»

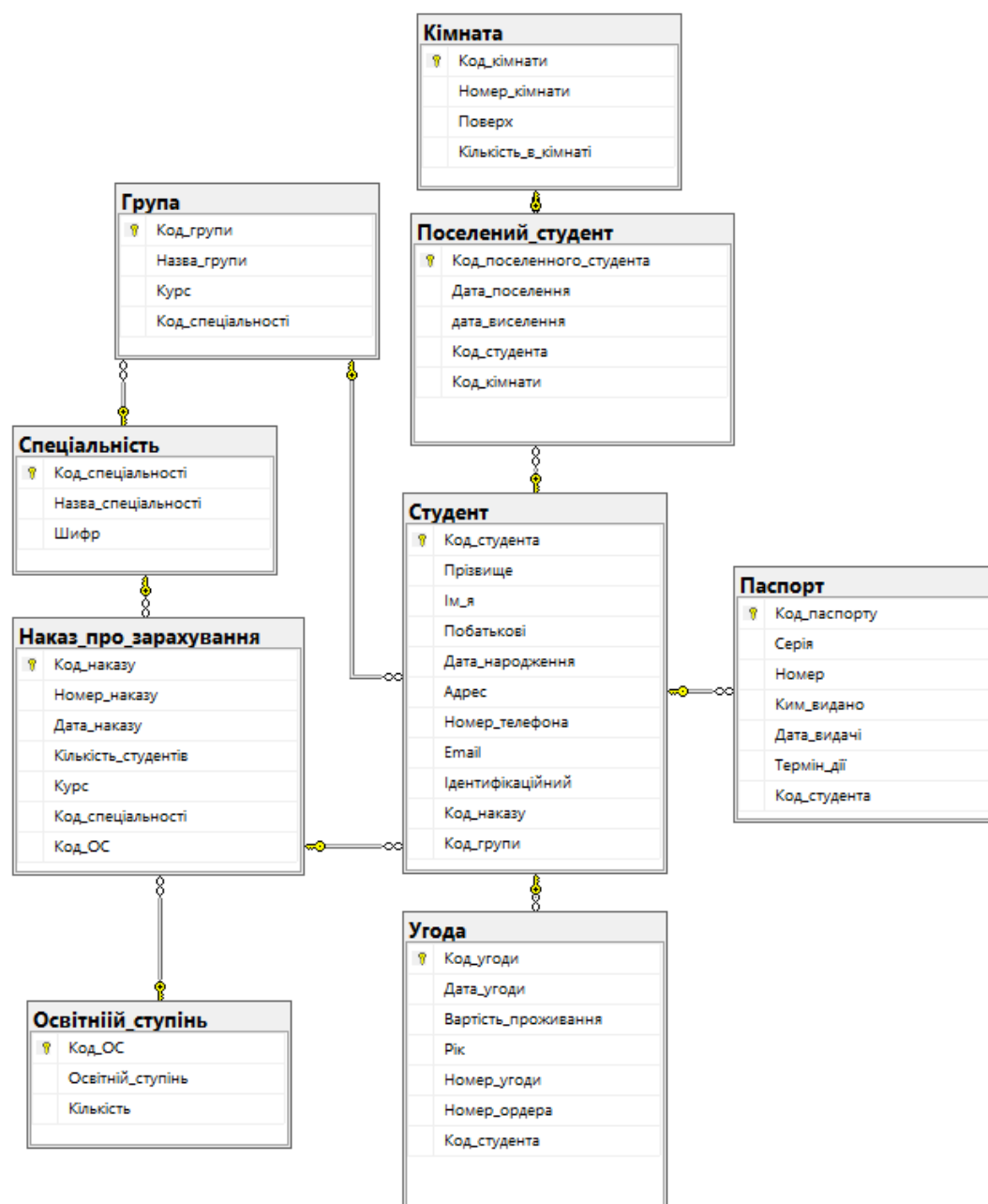


Рис.G.1 Схема вивантаженої бази даних в SQL 2018

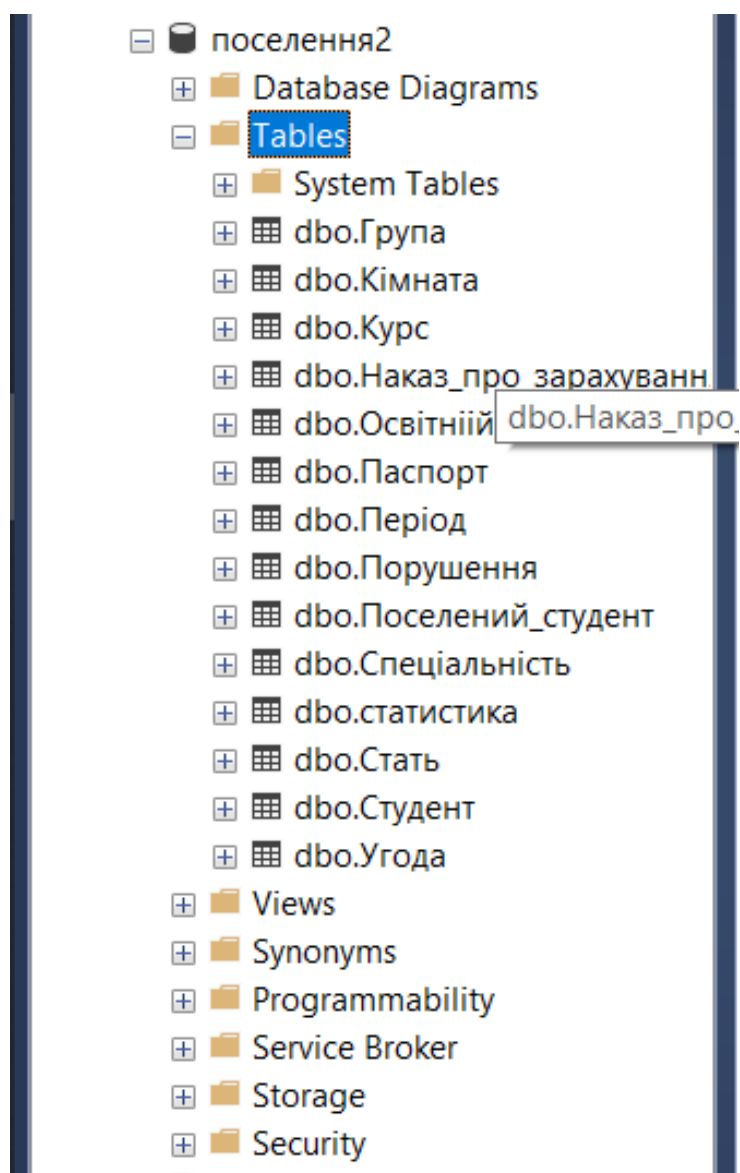


Рис.G.2 таблиці які потім були створенні під час роботи над проектом

ДОДАТОК «К»

Опис деяких фрагментів коду google sheets

```
var SS = SpreadsheetApp.openById('1oAIbcoS5LrI_H3XfTBNHLF-
K_xcwK2xaLQMThhEd5YY');
var sheet = SS.getSheetByName('dataFile');
var str = "";
```

Вище написано фрагмент коду для оголошення змінних, яким передається значення URL таблиць до яких буде підключений контролер esp8266

URL силка та назва файлу в таблицях Google позначені *жирним і курсивним шрифтом*.

Для підключення контролера ESP8266 до Google sheets використовуються такий фрагмент коду в Arduino IDE

```
const int chipSelect = 4;
SoftwareSerial Rfid = SoftwareSerial(2,3);
const char* host = "script.google.com";
const char *GScriptId = "1oAIbcoS5LrI_H3XfTBNHLF-K_xcwK2xaLQMThhEd5YY";
```

Підключіться до Wi-Fi і чекайте, поки з'єднання встановиться.

```
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
```

Запускає нове з'єднання HTTPS.

```
client = new HTTPSRedirect(httpsPort);
client->setInsecure();
client->setPrintResponseBody(true);
client->setContentTypeHeader("application/json");
```

Підключення до хостів. Ось це "script.google.com".

```
Serial.print("Connecting to ");
Serial.println(host);
```

Якщо клієнт підключений, то просто відправити дані в Google лист за допомогою функції POST.

```
if (client->POST(url2, host, payload)) {
    ;
}
else {
    ++error_count;
    DPRINT("Error-count while connecting: ");
    DPRINTLN(error_count);
}
```

Повний код на контролер ESP8266:

```
#include <SPI.h>                // Підключаємо бібліотеку SPI
#include <SD.h>                  // Підключаємо бібліотеку SD
#include <SoftwareSerial.h>
Sd2Card card;                  // вказуємо змінні
SdVolume volume;               // вказуємо змінні
SdFile root;                   // вказуємо змінні

const int chipSelect = 4;
SoftwareSerial Rfid = SoftwareSerial(2,3);
const char* host = "script.google.com";
const char *GScriptId = "1oAIbcoS5LrI_H3XfTBNHLF-K_xcwK2xaLQMThhEd5YY";

void setup() {
    Serial.println();
    Serial.print("Connecting to wifi: ");
    Serial.println(ssid);

    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());

    // Використовуйте клас HTTPSRedirect для створення нового TLS-з'єднання
    client = new HTTPSRedirect(httpsPort);
    client->setInsecure();
    client->setPrintResponseBody(true);
    client->setContentTypeHeader("application/json");
    Serial.print("Connecting to ");
    Serial.println(host);          // спроби підключитися до "script.google.com"
    Serial.begin(9600);           // Швидкість передачі 9600
    // з'єднання з сканером RFID
    Rfid.begin(9600);
```

```

while (!Serial) {}
Serial.print("\nInitializing SD card...");

if (!card.init(SPI_HALF_SPEED, chipSelect)) {
    Serial.println("initialization failed. Things to check:");
    Serial.println("* is a card inserted?");
    Serial.println("* is your wiring correct?");
    Serial.println("* did you change the chipSelect pin to match your shield or
module?");
    return;
} else {
    Serial.println("Wiring is correct and a card is present.");
}

// друкуємо тип карти
Serial.print("\nCard type: ");
switch (card.type()) {
    case SD_CARD_TYPE_SD1:
        Serial.println("SD1");
        break;
    case SD_CARD_TYPE_SD2:
        Serial.println("SD2");
        break;
    case SD_CARD_TYPE_SDHC:
        Serial.println("SDHC");
        break;
    default:
        Serial.println("Unknown");
}

// відкриття картки
if (!volume.init(card)) {
    Serial.println("Could not find FAT16/FAT32 partition.\nMake sure you've
formatted the card");
    return;
}
uint32_t volumesize;
Serial.print("\nVolume type is FAT");
Serial.println(volume.fatType(), DEC);
Serial.println();

volumesize = volume.blocksPerCluster();
volumesize *= volume.clusterCount();
volumesize *= 512;
Serial.print("Volume size (bytes): ");
Serial.println(volumesize);
Serial.print("Volume size (Kbytes): ");
volumesize /= 1024;
Serial.println(volumesize);
Serial.print("Volume size (Mbytes): ");
volumesize /= 1024;
Serial.println(volumesize);

```

```

    Serial.println("\nFiles found on the card (name, date and size in bytes): ");
    root.openRoot(volume);

// Читання всіх файлів
    root.ls(LS_R | LS_DATE | LS_SIZE);
}

void loop() {
    Serial.print("Humidity: "); Serial.print(h); //занесення значення в google
    sheets
    // перевірити, чи є якісь дані
    if(Rfid.available() > 0 ){
        // as long as there is data available...
        while(Rfid.available() > 0 ){
            // читання даних
            int r = Rfid.read();
            // роздрукувати його на монітор
            Serial.print(r, DEC);
            Serial.print(" ");
        }
        Serial.println();
    }
}

```