МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук,
управління та адміністрування
Кафедра автоматизованих систем
моніторингу навколишнього середовища

Бакалаврська кваліфікаційна робота

на тему: «Електронна розмітка та програмне кодування структурних елементів дипломних робіт за допомогою методології LaTeX»

Виконав: студент 4-го курсу групи К-41
спеціальність 122 – комп'ютерні науки
Нямцу Костянтин Євгенович

Керівник: к. т. н. , доцент
Великодний Станіслав Сергійович

Консультант: _____
_____

Рецензент: к. т. н. ,
Онищенко Сергій Михайлович

Одеса 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук, управління та адміністрування
Кафедра автоматизованих систем моніторингу навколишнього середовища
Рівень вищої освіти бакалавр
спеціальність 122 – комп'ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри АСМНС

_____

"13" квітня 2020 р.

З А В Д А Н Н Я
НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Нямцу Костянтину Євгеновичу
(прізвище, ім'я, по батькові)

1. Тема роботи «Електронна розмітка та програмне кодування структурних елементів дипломних робіт за допомогою методології TeX»

керівник роботи Великодний Станіслав Сергійович, к. т. н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від «13» квітня 2020 р. № 37-С

2. Строк подання студентом роботи 30 травня 2020 р.

3. Вихідні дані до роботи: система TeX; макропакет LaTeX; текстові процесори системи TeX; макроси по налаштуванню преамбули проекту; сукупність пакетів та команд зі сховища MiKeTeX; додаткові програмні забезпечення для роботи з OS Windows; зв'язка ПЗ з TeXStudio та MiKeTex; пакет шрифтів psyr; середовище розробки TeXStudio, Overlife.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): вступ; дослідження предметної області; історія розвитку системи TeX/LaTeX; переваги та недоліки системи TeX; редактори системи TeX/LaTeX; налаштування програмного забезпечення під OS Windows і Linux; можливості системи; структура документу в системі; структурні елементи системи; структура шаблону БКР; команди шаблону; висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): графічний матеріал виконаний у вигляді презентації

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання “13” квітня 2020 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Оцінка виконання етапу	
			у %	за 4-х бальною шкалою
1	Отримання завдання на роботу	13.04.2020		
2	Аналіз технічного завдання	16.04.2020		
3	Виконання аналітичного розділу	19.04.2020		
4	Виконання налаштувань для роботи	21.04.2020		
5	Дослідження структури системи	28.04.2020		
6	Дослідження методів опису об'єктів	03.05.2020		
7	Дослідження методів розбору шаблону	08.05.2020		
8	Проходження рубіжної атестації	11.05.2020		
9	Складання структури шаблону	18.05.2020		
10	Розробка і аналіз команд	23.05.2020		
11	Проведення тестувань фрагментів коду	27.05.2020		
12	Формування висновків до роботи	29.05.2020		
13	Перевірка роботи на плагіат	30.05.2020		
14	Складання графічної частини	02.06.2020		
15	Підготовка доповіді на захист	06.06.2020		
16	Проходження рецензування	08.06.2020		
	Інтегральна оцінка виконання етапів календарного плану (як середня по етапам)			

Студент _____ Нямцу К. Є.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Великодний С. С.
(підпис) (прізвище та ініціали)

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАК 5
ВСТУП 6
1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ 9
1.1 Історія розвитку системи TeX/LaTeX 10
1.2 Переваги та недоліки TeX/LaTeX 12
1.3 Редактори системи LaTeX 16
1.4 Налаштування ПЗ для роботи з OS Windows та Linux 22
2 СТРУКТУРА СИСТЕМИ TEX/LATEX 30
2.1 Можливості системи TeX/LaTeX 31
2.2 Структура документу в системі TeX/LaTeX 34
2.3 Структурні елементи системи 37
3 РОЗБІР ШАБЛОНУ БКР 39
3.1 Структура шаблону БКР 39
3.2 Команди, які використовувались при створенні БКР 40
ВИСНОВКИ 63
ПЕРЕЛІК ПОСИЛАНЬ 65
ДОДАТОК А 70
Фрагменти лістінгів TeX-коду експериментальної програмної реалізації 70
А.1 Вхідні налаштування преамбули у системі TeX 70
А.2 Завдання налаштувань преамбули для інтервалів виносок та шрифтів, а також для оформлення бібліографії 70
А.3 Завдання налаштувань для списків та нумерації сторінок 71
А.4 Завдання налаштувань для списків та нумерації сторінок 72
А.5 Завдання налаштувань для заголовків 73
А.6 Завдання налаштувань для оголошення 73

ПЕРЕЛІК УМОВНИХ ПОЗНАК

MS – Microsoft;

БД – База даних;

БКР – бакалаврська кваліфікаційна робота;

ДСТУ – Державний стандарт;

ОС – Операційна система;

ПЗ – Програмне забезпечення;

СНД – Співдружність Незалежних Держав;

UML – Unified Modeling Language;

WEB – World Wide Web;

WYSIWYG – What You See Is What You Get

ВСТУП

На сьогоднішній день, макропакет LaTeX є дуже популярним у всьому світі. Найбільшого розповсюдження система TeX/LaTeX досягла в західних країнах. Це обумовлено декількома факторами: по – перше, LaTeX є дуже потужним інструментом для роботи з текстом, тому що він включає в себе тисячі пакетів та десятки тисяч команд, які потрібні для проведення різного роду операцій над документом, по – друге, система являється безкоштовною та багатоплатформовою, це означає, що її дистрибутиви знаходяться у вільному доступі, та мають відкритий код, а саме тому і постійно розвиваються не тільки зі сторони розробників, а ще й за допомогою користувачів. Якщо ж наприклад взяти країни СНД, та проаналізувати кількість людей, які користуються TeX/LaTeX, то вона не перебільшить 7% від загальної кількості. В Україні система почала розвиватись з 2017 року, однак цей процес відбувається дуже повільно і на 2020 рік тільки 4% від усіх користувачів віддають перевагу LaTeX.

Слід зазначити, що на момент створення цієї БКР, на просторах всесвітньої павутини не було жодного посібника на державній мові, по роботі з даною системою. Підставою для виконання БКР послужив факт того, що кожен рік студенти оформлюють бакалаврську кваліфікаційну роботу згідно з ДСТУ. Не для кого не є таємницею, що на цей процес витрачається дуже багато сил, та часу, які студент міг би направити на більш поглиблене вивчення наукового питання, яке ставиться перед ним темою його роботи. Для вирішення цієї проблеми, ми можемо звернутись за допомогою до системи TeX/LaTeX. Саме вона дозволяє своїм користувачам зосередитися на наповнені документу контентом, а не витрачати купу часу на його оформлення, та дизайн. Все це є можливим, завдяки особливостям структури системи, яка дозволяє користувачам, не тільки верстати документи найвищої якості, а ще й робити шаблони. Шаблони у LaTeX, дозволяють максимально швидко отримувати готову, унікальну роботу, яка буде оформлена відповідно

до заданих налаштувань у преамбулі документу. Ці налаштування задаються самим користувачем. Треба відзначити, що при використанні шаблона, user'у потрібно тільки замінити інформаційну частину, на ту яка йому потрібна. На виході він отримає вже свій проект. Саме цей підхід і надає системі ту гнучкість, завдяки якої вона так популярна. Тому виходячи з цих фактів, можна підкреслити актуальність по створенню загального шаблону для дипломного проекту.

Метою даної роботи є формування програмного шаблону коду розмітки бакалаврської кваліфікаційної роботи (БКР) за допомогою мови LaTeX.

Для досягнення цієї мети потрібно виконати наступні задачі:

- проаналізувати структурні елементи БКР на предмет виділення компонентів, що підлягають переведення до шаблону;
- виділити характерні ознаки структурних елементів БКР та подати їх за допомогою методології UML;
- виконати аналіз засобів та середовищ підтримки програмування у форматі *.tex;
- виконати формування файлу розмітки за допомогою методології LaTeX;
- проаналізувати отримані експериментальні результати на предмет відповідності вимогам щодо оформлення БКР.

Кожна частина роботи виконана згідно з ДСТУ та містить пояснення щодо її кодування. Весь код розбитий на частки згідно з структурою БКР, та містить максимально чітке тлумачення. Для створення цієї роботи було використано більш ніж 20 навчальних посібників, більшість з яких написана на англійській мові, вся інформація була аналізована та перевірена на практиці у LaTeX редакторі TeXStudio.

Робота складається з 3-х розділів. В першому розділі описується історія створення TeX/LaTeX, а також розглядаються всі плюси та мінуси системи TeX/LaTeX, додаткове ПЗ для роботи з системою, та процес його установки

на ОС Windows і Linux. Другий розділ присвячений розгляду основних пакетів і команд, які розкривають повну структуру системи TeX/ LaTeX. Третій розділ описує саме роботу з шаблоном для БКР, розглядаються окремі команди, які дозволяють оформлювати дипломний проект згідно з вимогами.

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

LaTeX за своєю суттю є нічим іншим, як складанням розширень макро рівня, які були зібранні з однією важливою метою, яка полягає в доповненні мови комп'ютерної верстки TeX. Його найчастіше порівнюють з різними елементами, які тим чи іншим чином відносяться до будь якої редакторської системи, але найправильнішим буде його порівняння з програмним забезпеченням [1] [1)]. Цей висновок виходить саме з того, як влаштована робота в даній системі. Якщо докладніше сам користувач, за допомогою цілого каталогу команд, вчить систему різноманітним тонкощам для облегшення своєї ж роботи. Слід зазначити, що TeX/LaTeX – це дуже вдалий вибір зі сторони рядового користувача, тому що навчивши систему всього одного разу, вона запам'ятає, як саме оформлюються документи цього типу, тобто створить деякий шаблон документу. Наступного разу коли користувачу знадобиться зверстати такий само документ, він може звернутися до вже існуючого шаблону. Після цього йому треба буде замінити вхідну інформацію в документі, для того щоб отримати новий результат. Якщо користувачу потрібно наприклад зробити іншу наукову або письменницьку роботу, він має можливість або зробити новий шаблон, або скачати його в Інтернеті. Саме в мережі можливо знайти шаблон документу любого типу, але це зовсім не вигідно, тому, що ці документи оформлені інакше, а перероблювати преамбулу вже готового проекту, може бути дуже складно [2] [2)]. Набагато легше написати код для власної преамбули, який буде задовольняти всі правила оформлення, та ДСТУ, які потрібні користувачу.

Людина, яка вперше зустрілась з LaTeX і не має досвіду у програмуванні, скоріш за все не зрозуміє принципів його роботи. Щоб

[1)] [1] Kopka H., W. Daly P. A guide to LaTeX: document preparation for beginners and advanced users (навч. посібник). 3rd ed. p. cm. London: Addison – Wesley, 1999. 609p.

[2)] [2] Великодний С. С., Нямцу К. Є. Сучасна інформаційна система дляпідготовки LaTeX-документів «TeXstudio». Інформатика, інформаційнісистеми та технології: тези доп. 16-ї Всеукр. конф. студ. і мол. наук. Одеса, 19 квітня 2019 р. Одеса, 2019. С. 11–13.

оволодіти навичкою верстання в даній системі потрібен деякий досвід, який приходить з часом. Якщо користувач раніше вже займався версткою наприклад сайтів, для нього освоєння команд системи не буде важким.

1.1 Історія розвитку системи TeX/LaTeX

Система комп'ютерної верстки TeX, являється революційною розробкою в області типографії. Вона була розроблена видатним американським вченим в області інформатики, ім'я якого Дональд Кнут. TeX було розроблено цілком на мові програмування SAIL [3] [1)]. Ідея створення даної редакторської системи прийшла до Кнута в час коли печатні станки, змінювались персональними комп'ютерами. Тоді зовсім не було достойних редакторів, саме тому, текст який набирали користувачі ПК, був громіздкий, а формули набрані на клавіатурі було зовсім не можливо розібрати. Кнут вирішив створити таку систему комп'ютерної типографії, яка надала б людям можливість набирати текст не гірший ніж на печатному станку, сидячи дома у монітора.

Перша версія TeX була розроблена у 1979 році , та отримала признання в різних областях науки. Особливо сильно система спростила життя математичним факультетам та математикам того часу в цілому. Однак TeX тієї версії була не ідеальна. Вона дозволяла виконувати свою роботу науковим діячам, але для простих користувачів, які не мали відношення до точних наук, розібратись з інтерфейсом було дуже складно. Дана система включала на той час кількість команд, яка у сотні разів менша за ту, що містить система TeX/LaTeX на сьогоднішній день. Кнут добре усвідомлював, що TeX не є досконалою, саме тому він продовжував придумувати і впроваджувати в систему нові алгоритми, та пакети.

[1)] [3] Goossens M., Mittelbach F, Braams J., Carlisle D., Rowley C. The LaTeX Companion (навч. посібник). 2nd ed.p.cm. Boston: Addison–Wesley, 2004. 1119p.

Нова, переписана з нуля версія TeX, була видана в 1982 році і названа TeX82. Крім інших змін початковий алгоритм переносів був замінений новим. Щоб забезпечити відтворюваність результату на різному устаткуванні, замість арифметики з плаваючою комою TeX82 використовував арифметику з фіксованою комою.

Вже 1989 році Кнут знову перезапускає систему з нуля, та випускає на додачу, ще й новітнє програмне забезпечення – METAFONT, яке було розроблено для роботи з векторними шрифтами, які ще більше доповнювало нову систему. На цей раз система дозволила користувачам використовувати аж 256 символів, замість 128 звичних. З версії 3.0 TeX використовує оригінальну систему нумерації версій: кожне оновлення додає додаткову десяткову цифру в кінці номера версії так, що вона постійно асимптотично наближається до числа π [4] [1)].

Не зважаючи увагу на всі спроби Дональда Кнута, спростити роботу у системі TeX для рядових користувачів йому так і не вдалось. Набір та верстка документів перетворювались на чистого виду написання коду на мові програмування, саме тому система спочатку була не дуже популярною. Однак у 1984 році, через два роки після першого оновлення TeX, видатним діячем в області інформатики, Лєсли Лемпортом, був розроблений макропакет LaTeX. Він мав можливість підключитись до системи комп’ютерної верстки TeX, та автоматизувати її процеси. Тепер для роботи, звичайному користувачу треба було тільки підключити потрібні пакети, та знати команди, які ці пакети надають. Слід звернути увагу, що після підключення LaTeX, система заграла новими фарбами, та почала набирати свою популярність. Тим самим було зроблено дуже великий крок у розвитку комп’ютерної типографії.

Перша версія LaTeX мала багато недоробок, тому вже у 1994 році на просторах інтернету, з’являється нова версія макропакету, з культовим ім’ям

[1)] [4] Kottwitz S. Latex Beginer’s Guide (навч. посібник). Olton: Packt Publishing, 2011. 336p.

LaTeX2ε. В ній були виправлені помилки попередньої версії. С того часу розробники LaTeX, кожного року випускають нові оновлення. На сьогоднішній день для того, щоб почати роботу з LaTeX, не потрібно його нікуди підключати, все це за користувача роблять додаткові програмні забезпечення, які просто завантажуються з WEB. Розробка системи TeX зупинилась у березні 2008 року і тепер всі обновлення взяли на себе розробники макропакету LaTeX.

Незабаром очікується вихід нової версії макропакету, зі слів творців стає зрозуміло, що оновлення буде глобальним і впровадить у систему багато новітніх розробок та алгоритмів, що ще більш полегшить роботу з TeX/LaTeX.

1.2 Переваги та недоліки TeX/LaTeX

Як вже було згадано, LaTeX є дуже потужним інструментом для верстки різного типу документації. Система TeX/LaTeX має велику кількість корисних властивостей, які у порівнянні з текстовими редакторами пакету MS Office, надають користувачу значущі переваги у роботі з текстом та його оформленні (рис.1).

Розглянемо більш детально кожен елемент діаграми приведеної нище. Першою, та дуже важливою перевагою системи TeX/LaTeX над іншими редакційними системами, є його можливість змінювати оформлення документу використовуючи всього одну команду з параметрами. Незалежно від обраного типу документу, користувач завжди має доступ до зміни класу документу. Наприклад user може змінити клас зі статті на книгу. Тоді, коли над переробленим кодом буде проведена компіляція, система повністю перестроїться під нові налаштування, і почне оформлювати документ згідно зі стандартами системи, які користувач за бажанням може змінити коли завгодно. Далі слід розповісти про таку особливість системи, як багатоплатформовість. Вона полягає у тому, що LaTeX може компілювати

файли з поширенням .tex, у такі формати, як DVI та PDF. Це дає змогу користувачам, запускати ці файли на різних системах, виконувати над ними маніпуляції, а також печатати документ на будь якому принтері.

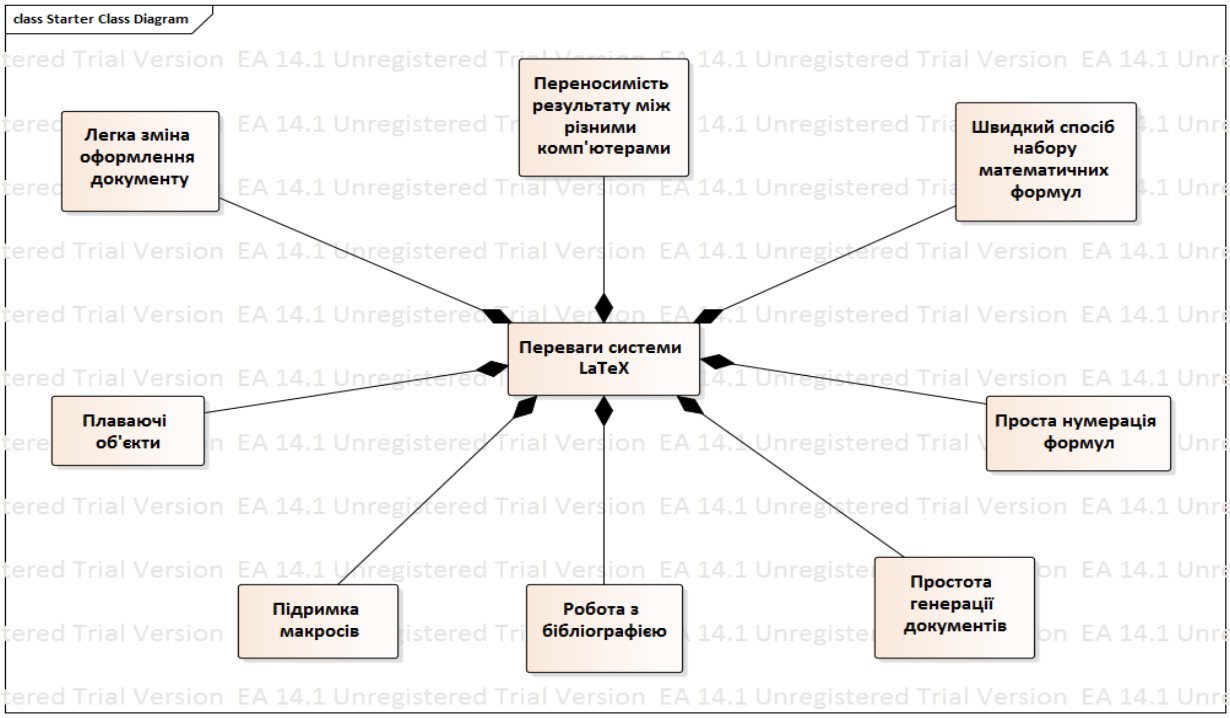


Рисунок 1 – Переваги макропакету LaTeX

Ще одним плюсом, є те, що система дозволяє печатати формули високої якості, без використання додаткових ПЗ [5] [1)]. Користувач може сам вибрати стиль шрифту формул із стандартного списку шрифтів для LaTeX. Це дає перевагу над редакторами типу WISIWYG, в яких людині потрібно витрачати багато часу, для того, щоб подати формулу у стилі, якому вимагають правила оформлення наукових робіт різного типу. Також, макропакет дозволяє робити зручні посилання за допомоги всього пари команд, які будуть розглядатись у другій частині даної БКР. LaTeX дозволяє не тільки працювати зі списками джерел а й працювати з бібліографічними базами даних. В такому випадку повністю автоматизується не тільки процес стеження за нумерацією, але і оформлення бібліографії згідно зі стандартами.

[1)] [5] Роженко А. Искусство верстки в LaTeX'е (навч. посібник). Новосибирск: Изд. ИВМиМГ СО РАН, 2005. 398 с.

Швидкість роботи системи LaTeX, набагато вище, ніж в тому ж MS Word, оскільки LaTeX дозволяє своїм користувачам не зациклюватися над дизайном, оформлення тексту проходить автоматично, згідно з налаштуваннями на початку документу. Користувачеві доводиться обмірковувати тільки контентну частину роботи, це дає йому змогу повністю зосередитись на виконані свого завдання.

Більшість редакторів LaTeX, є редакторами з відкритим вихідним кодом, а як ми знаємо, програмні забезпечення з відкритим кодом натякають на безкоштовність [6] [1)]. Також LaTeX підтримує написання тексту на абсолютно будь-якій мові, в рамках документу. Ще система підтримує макроси дозволяють створювати новітні команди, котрі можуть навіть перекривати властивості стандартних команд. Останнім плюсом LaTeX, є велика кількість пакетів завдяки яким користувач має майже безмежні можливості оформлення та редагування тексту. Ці пакети можливо підключити тільки за допомогою спеціального ПЗ, яке дозволяє завантажувати їх прямо з інтернету.

Крім переваг, система TeX/LaTeX має декілька неважливих недоліків, які для деяких користувачів можуть показатися навіть позитивними (рис.2). Основним недоліком системи є складність конвертації документів з розширенням .tex, на приклад у таке популярне розширення, як .doc [7] [2)]. Це питання звісно можливо вирішити, однак процедура може зайняти забагато часу. Система TeX/LaTeX досить складна в освоєнні, і перед роботою необхідно ознайомиться з нею краще, в спеціальних посібниках. Також не є системою типу WYSIWYG (що бачиш, то і отримаєш). Це пов'язано з тим, що при написанні коду, ми не можемо бачити текст до компіляції, на відміну від Word.

[1)] [6] Балдин Е. Компьютерная типография LaTeX (навч. посібник), Санкт-Петербург: БХВ Петербург, 2008. 304 с.

[2)] [7] LaTeX – Wikipedia (заголовок з екрану). URL: https://ru.wikipedia.org/wiki/ LaTeX (дата звернення: 19.04.2020).

Слід також зазначити, що все оформлення набирається за допомогою спеціального коду, який складається з команд котрі керують версткою документів, та підключенням додаткових пакетів для розширення функціоналу. Спершись на цій факт виходить, що користувачу для якісного оформлення свого документу чи проекту, потрібно мати під рукою каталог з командами, тому що синтаксис усього коду дуже складно запам'ятати, навіть досвідченому користувачу [8] [1)].

Система не призначена для швидкої верстки складних документів, то му що розробники більше орієнтувались не на швидкість виконання, а на якість роботи на виході. Однак з цього недоліку вилазить вагомий плюс системи, який полягає у якісній структурі елементів, та їх розміщення щодо один одного. Такою системою дуже легко керувати, тому що вона є дуже гнучкою та мобільною.

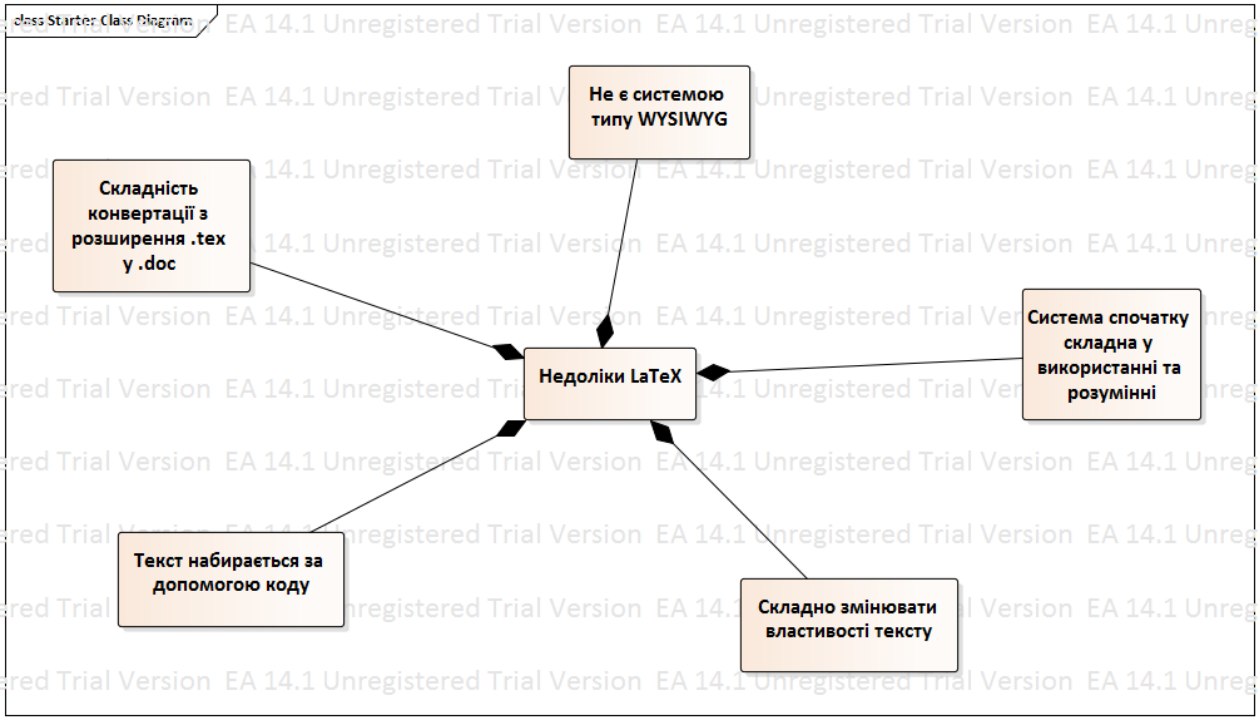


Рисунок 2 – Недоліки макропакету LaTeX

[1)] [8] Microsoft Word та Latex: снова вместе (заголовок з екрану). URL: https://https://itc.ua/articles/microsoft_word_i_latex_teper_vmeste_27030 (дата звернення: 19.04.2020).

Виходячи з наведеної інформації, можна зробити висновок, що система LaTeX, має недоліків менше ніж переваг. Саме це є добрим знаком, для того щоб почати користуватись даною системою. Якщо розглядати її недоліки під певним кутом, можна зрозуміти, що ці недоліки, є її сильною стороною.

1.3 Редактори системи LaTeX

Макропакет LaTeX дає можливість перебудовувати вихідний текст в дуже якісну продукцію для друку, однак для створення такого тексту необхідне втручання додаткового програмного продукту – такі продукти називаються редакторами LaTeX. Дані програмні засоби використовуються не тільки для створення файлів LaTeX, але також і для здійснення різних видів маніпуляцій з текстом (наприклад програмування або створення сценарію для тексту). Звичайно для набору тексту документа можна використовувати навіть найпростіші редактори, однак якщо підібрати правильний редактор для системи LaTeX, то працювати стане набагато зручніше, а головне покращиться якість роботи що є дуже важливим фактором.

Було проаналізовано більш десяти редакторів LaTeX, та відібрані ті які мають максимально зручний інтерфейс, і включають у себе новітні технології для верстки документів. Редактори, які описуються нижче – є найкращим вибіром, для сучасного користувача, так як база цих ПЗ постійно доробляється та оновлюється.

Першим в нашому списку редакторів з відкритим початковим кодом [9] [1)] йде Open LaTeX Studio (рис.3). Його особливістю являється те, що він дає можливість користувачам працювати над документом спільно, онлайн, а це значить, що припустимо, якщо над документом працює певна команда

[1)] [9] Великодний С. С., Тимофєєва О. С., Зайцева-Великодна С. С., НямцуК. Є. Порівняльний аналіз властивостей відкритого, вільного такомерційного програмного забезпечення. Інформаційні технології такомп’ютерна інженерія. 2018. № 1 (41). С. 21–27.

людей по Інтернету, то їм не доведеться купувати, та завантажувати додаткові ПЗ, що б працювати разом. Слід зазначити що інтерфейс в даному редакторі побудований на базі NetBeans (платформа, яка сприяє зручному та простому освоєнню інтерфейсу). Оскільки це програмне забезпечення створено з використанням ідеї відкритого вихідного коду, то будь-який бажаючий користувач може, як запропонувати нові оновлення для поліпшення роботи, так і зробити їх сам. Також дуже важливим фактором, який обумовлює популярність цього програмного забезпечення є його простий дизайн, який дуже легко сприймається будь-якою категорією людей, незалежно від їх рівня знань в системі.

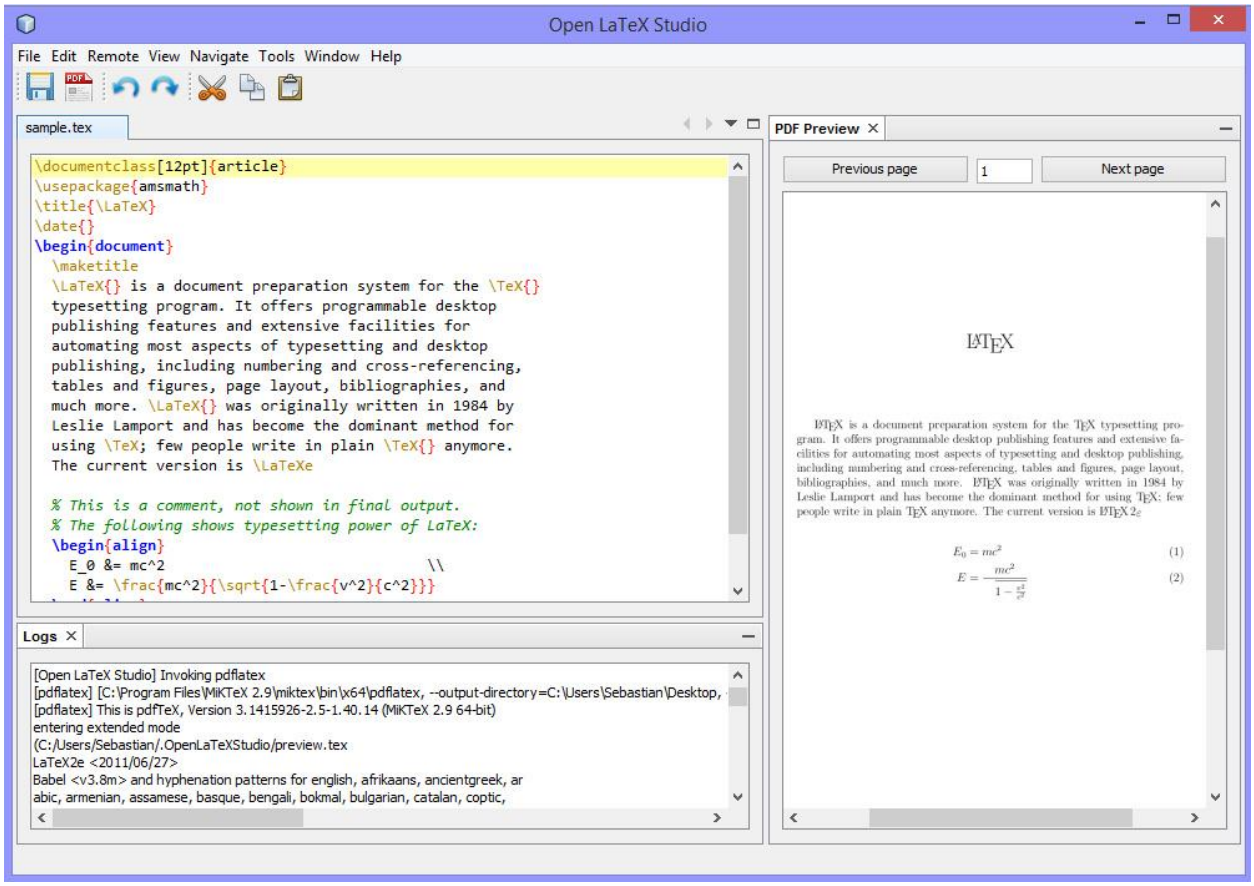


Рисунок 3 – Інтерфейс редактора Open LaTeX Studio

Overleaf є онлайн редактором (рис. 4). Оскільки він також має відкритий вихідний код, є можливість працювати з ним на будь – якій

платформі [10] [1]. З його плюсів слід відзначити наявність великої кількості навчальної документації, а так само можливість створювати безкоштовні проекти в необмеженій кількості. Включає в себе такі компілятори як latex, pdflatex, а також має функціонал не гірше інших редакторів. Також слід зазначити, що Overleaf надає своїм користувачам можливість працювати з одним документом командою. Це дуже гарне доповнення для тих людей, які роблять проект у групі.

Цей редактор є дуже популярним у західних країнах і дуже багато користувачів обирають саме його, через його гнучкість, та не прихильність до певної системи, чи процесору.

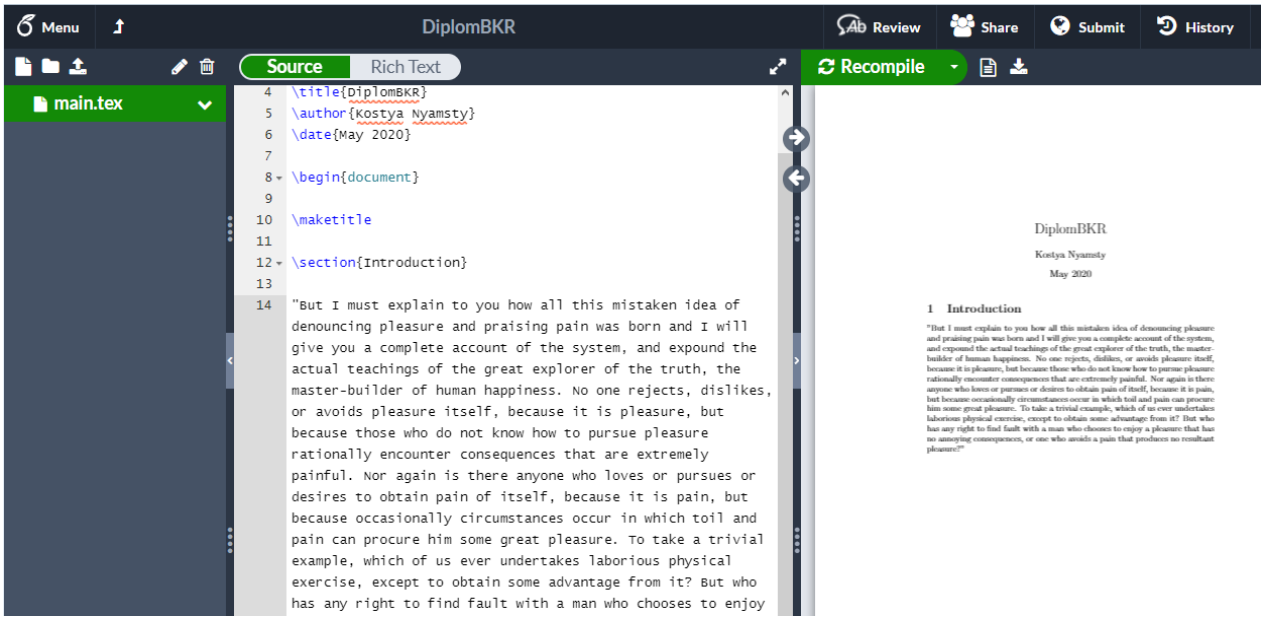


Рисунок 4 – Інтерфейс редактору Overleaf

Texmaker є одним з кращих редакторів LaTeX (рис.5). Його переваги полягають в тому, що він по – перше підходить під будь-яку операційну систему (що є досить важливим фактором), по – друге легко освоюється користувачами, будь то просунутий або недосвідчений користувач. Також слід

[1] [10] ShareLaTeX, Онлайн редактор LaTeX (заголовок з екрану). URL: https://ru.sharelatex.com (дата звернення: 20.04.2020).

зазначити що в Texmaker існує багато зручних інструментів для редакції тексту, наприклад автоматичне завершення коду, підсвічування синтаксису або перевірка орфографії. Слід згадати також, що є можливість обробки коду та тексту різними типами компіляторів, залежно від потреби користувача при створенню свого проекту. Наприклад самими популярними серед користувачів є компілятори: PdfLateX, XeLateX, Makeindex, BiBTex та багато інших. Саме тому редактор Texmaker має своїх прихильників, яких приваблює можливість системи, компілювати документи в інші різноманітні розширення. Ця властивість надає редактору певної гнучкості.

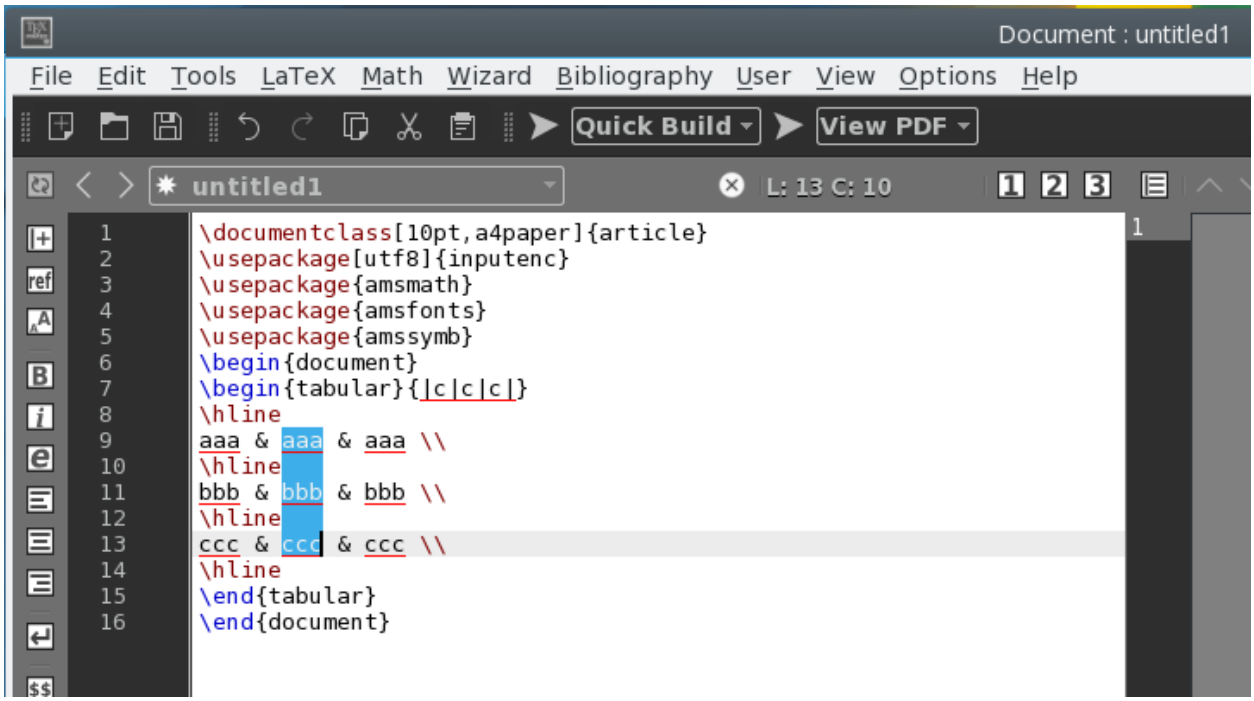
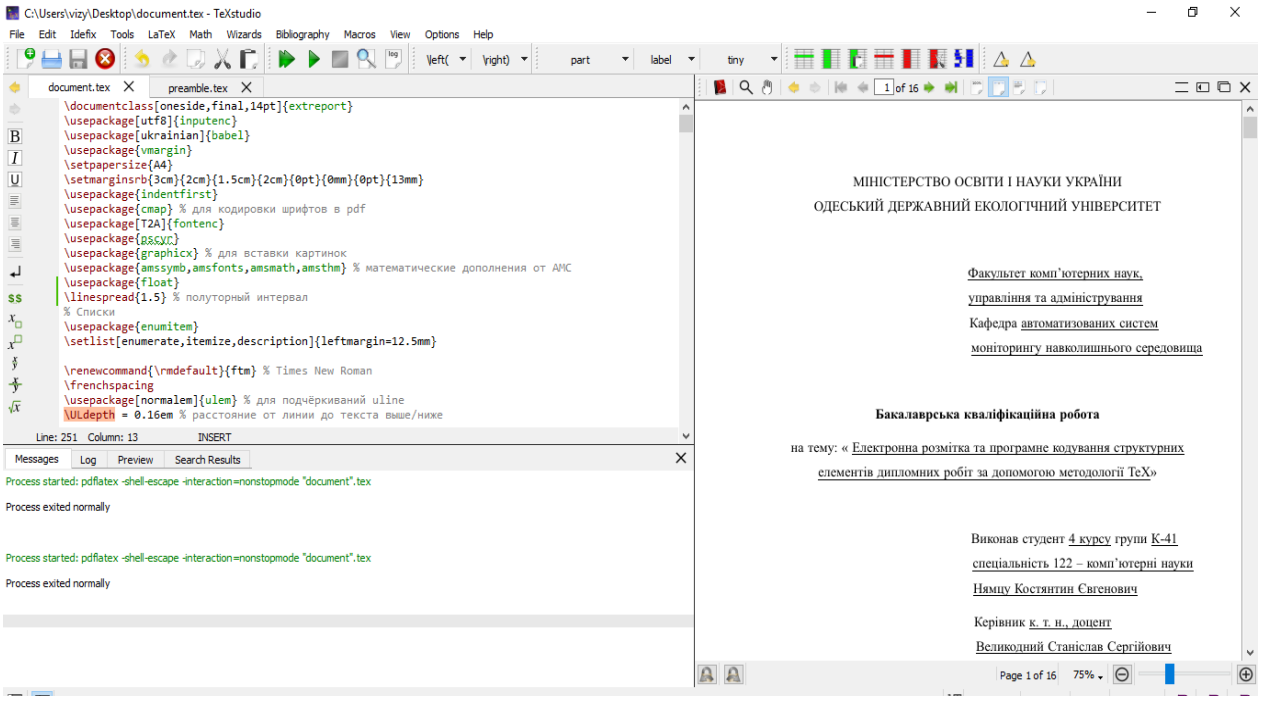


Рисунок 5 – Інтерфейс редактора Texmaker

TeXstudio – зручний графічний редактор для роботи з LaTeX (рис.6), який дає можливість легко створити і редагувати потрібний документ в одному з форматів TeX / LaTeX. TeXstudio створений з метою максимального спрощення роботи з документацією LaTeX, а також для

поліпшення якості друку. Розробники даного програмного засобу [11] [1)] подбали про своїх користувачів і надали величезну кількість інструментів для створення і оформлення текстових документів. TeXstudio повністю підтримує Unicode [12] [2)]. Функції TeXstudio дозволяють управляти скриптами користувачів, аналізувати текст, підраховувати і визначати, як часто в тексті з'являються нові слова і пропозиції. За допомогою навігатора користувач може переходити до вкладок і виділеним рядкам або помилок. Саме ці його можливості надають TeXstudio перевагу над іншими редакторами. А покращений перегляд структурного документа надає можливість ознайомитись з логічної ієрархією файлів, а перегляд формул часто відображається у верстці LaTeX. Із скомпільованих у PostScript LaTeX-файлів можливий експорт до HTML, також вбудований переглядач PDF-файлів.



[1)] [11] Великодний С. С., Тимофєєва О. С., Зайцева-Великодна С. С. Метод розрахунку показників оцінки проекту при виконанні реінжинірингу програмних систем. Радіоелектроніка, інформатика, управління. 2018. №4. С. 135–142. DOI: 10.15588/1607-3274-2018-4-13.

[2)] [12] Обзор редакторов для работы в LaTeX (заголовок з екрану). URl: http://mydebianblog.com/2013/02/latex-editors-aintegratedlatex.html (дата звернення: 23.09.2019).

Рисунок 6 – Інтерфейс редактору TeXstudio

Atom редактор (рис.7), такий же універсальний редактор LaTeX системи з відкритим кодом, як і попередні, його відмінність полягає в його сучасному інтерфейсі, створеному на основі веб технологій, які є дуже популярними на сьогодні. Також у даного редактора є ціла система пакетів, в якій велику частину представляють розширення функціональності.

За функціоналом в цілому йому притаманні всі можливості, які повинні бути в редакторі. Atom є дуже потужним інструментом тому, що дає можливість своїм користувачам працювати не тільки з LaTeX, але й писати наприклад програмний код, який відповідає наприклад за WEB програмування і роботу у браузері. Також це програмне забезпечення допускає колаборації системи LaTeX з іншими популярними системами із функціями тегування тексту. Можливо об’єднання наприклад з HTML або XHTML кодом, для досягнення певної мети. Ще можна відмітити швидкість обробітки даних при компіляції, навіть на досить старих PC.

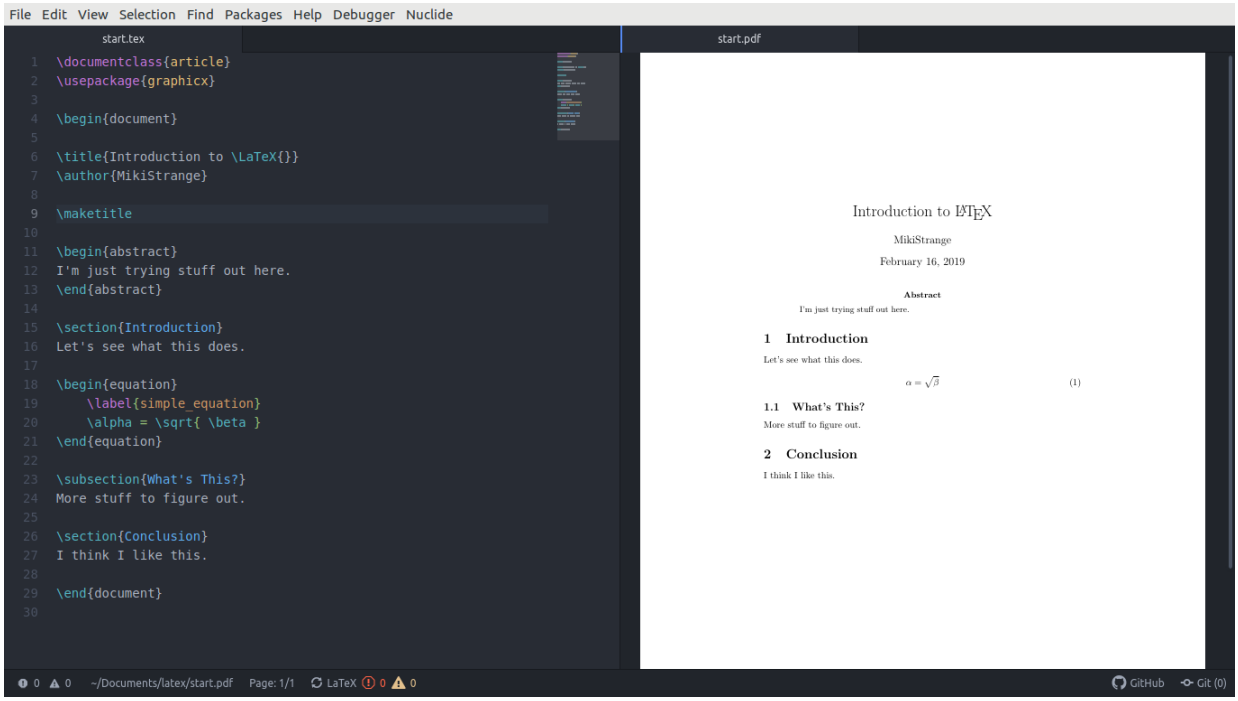


Рисунок 7 – Інтерфейс редактору Atom

Виходячи з перерахованої інформації можна зробити висновок, що існує величезна кількість LaTeX редакторів, як з відкритим кодом, так і не відкритим. З даного достатку якісного програмного продукту [13] [1)], можна вибрати абсолютно будь - який редактор, який задовольняє потреби рядового користувача. Починати своє навчання бажано з простих редакторів, з легким і зрозумілим інтерфейсом, а потім переходити на більш складні, які включають більше різних функцій для поліпшення якості роботи. LaTeX це вибір на користь прогресу. Нове покоління редакторів користується великою популярністю на Заході, і я вважаю, що зовсім скоро вони повністю замінять нам найпростіші текстові редактори. Завдяки цьому ходу, буде по – перше виключено піратство, а по – друге впроваджено нові технології, яка зможуть замінити пакет MS Office.

1.4 Налаштування ПЗ для роботи з OS Windows та Linux

Windows і Linux є дуже популярними системами на сьогоднішній день. Саме тому нижче буде приведена інструкція, по установці додаткового ПЗ, яке дозволить підключати пакети з Інтернету, та працювати з системою TeX/LaTeX у цілому.

Додаткове ПЗ без якого не можливо буде запустити LaTeX на Windows, носить назву MikTeX. Само ПЗ завантажити дуже легко. Для цього користувачу потрібно перейти на офіціальний сайт MikTeX (рис.8), за допомогою цього посилання – https://miktex.org/download.

[1)] [13] Великодний С. С. Метод представлення оцінки реінжинірингупрограмних систем за допомогою проектних коефіцієнтів. Сучасний станнаукових досліджень та технологій в промисловості. 2019. № 1 (7). С. 34–42. (кат. «Б») DOI: https://doi.org/10.30837/2522-9818.2019.7.034.

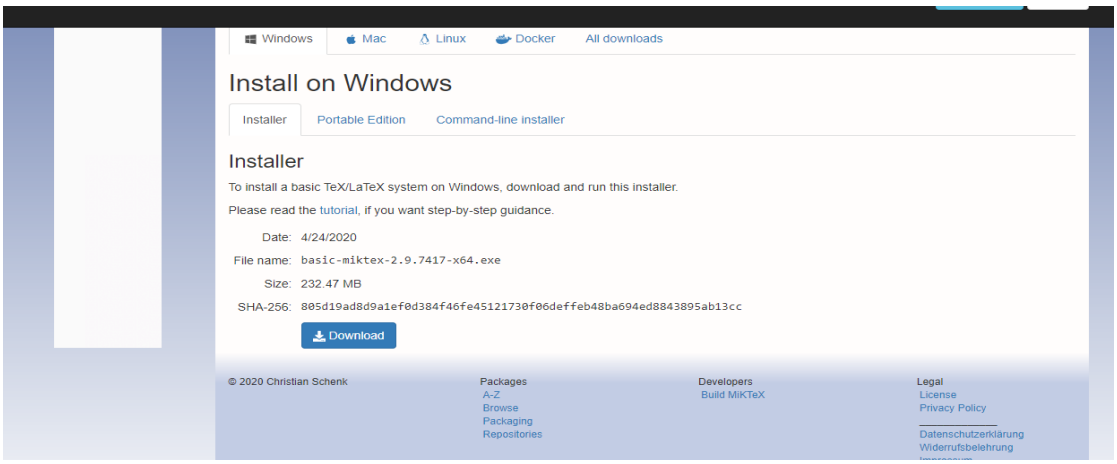


Рисунок 8 – Офіціальний сайт MikTeX

Потім натиснути на кнопку Dowload, та загрузити програмне забезпечення на власний ПК. Далі потрібно обрати будь який редактор з перерахованих у пункті1.3. Для прикладу був взятий редактор TeXstudio, через його можливості, та безмірну популярність серед юзерів. Завантажити його можна перейшовши по посиланню – https://www.texstudio.org (рис.9).

На мою думку, якщо мова йде про розробку проекту [14] [1)] на ОС Windows, то найкращою зв’язкою є сукупність таких програмних забезпечень, як TeXstudio та MikeTex, тому що вони дуже гнучкі у використанні, та дозволяють користувачам, при бажанні досягти оформлення найвищого ступеня складності оформлення, структурного документу. Саме тому ця зв’язка ПЗ є дуже популярною серед усіх користувачів.

[1)] [14] Великодний С. С. Методи реінжинірингу програмних систем. Технологии приборостроения. 2014. Спец. вып. С. 65–68.

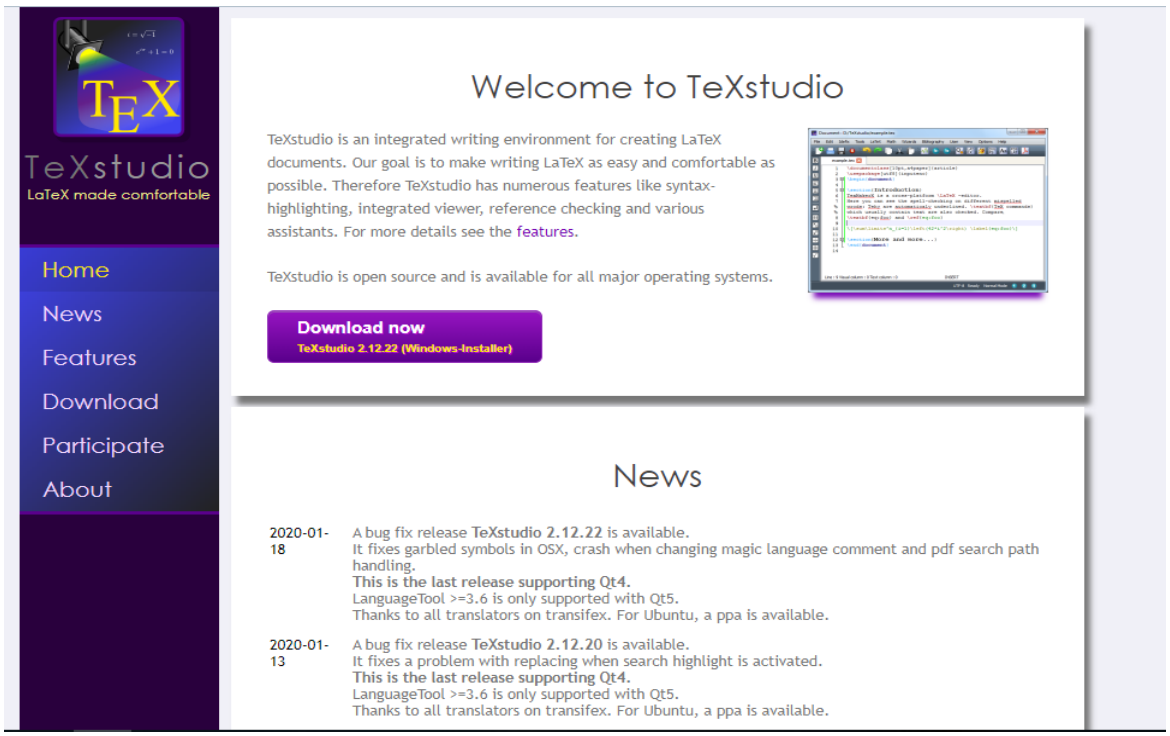


Рисунок 9 – Офіціальний сайт TeXstudio

Інтерфейс TeXstudio було розглянуто у пункті 1.3. А інтерфейс MikeTeX 2020 року виглядає так (рис.10). Він є дуже зручним та зрозумілим будь якому користувачу, який хоча б раз мав справу з програмним забезпеченням, створеним під ОС Windows. Він може працювати у режимі перегляду, а та у режимі адміністрування (завдяки якому користувач може налаштовувати MikeTeX виходячи з власних потреб). Після завантаження даного ПЗ потрібно буде докачати пакети з його дистрибутиву. Це основні пакети для верстки у системі, однак, якщо у преамбулі документа будуть підключенні ще якісь пакети котрі не були завантажені с першого разу, то MikeTeX налаштує їх прямо під час роботи [15] [1)].

[1)] [15] Установка и настройка программ для редактирования LaTeX файлов (заголовок з екрану).URL: http://blog.harrix.org/article/849 (дата звернення: 23.04.2020).

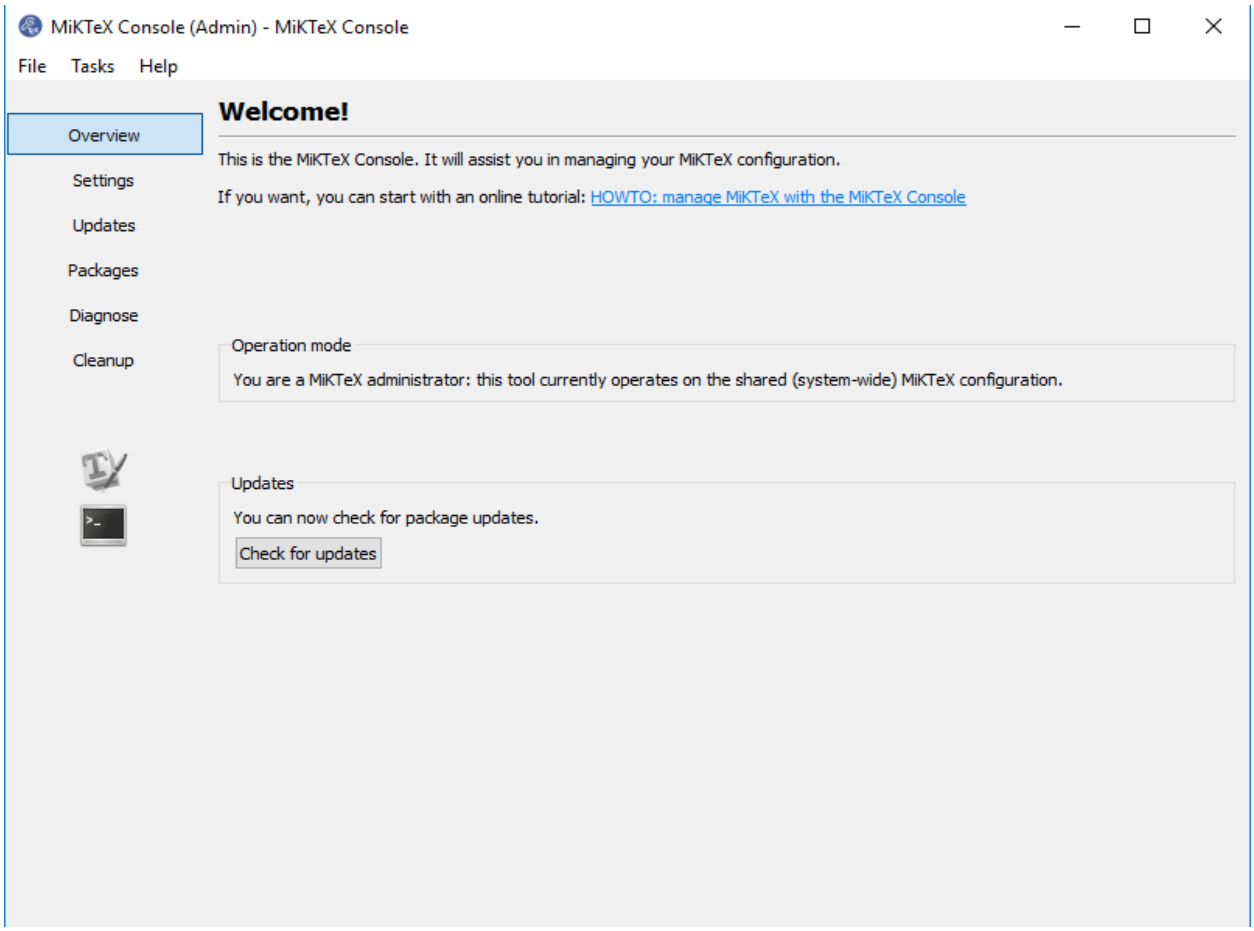


Рисунок 10 – Інтерфейс ПЗ MikeTeX

Найважчим етапом у налаштуванні ПЗ для роботи на Windows, є підключення додаткових шрифтів. Вони подаються у пакеті, однак для того щоб їх налаштувати, потрібно змінити деякі властивості MikeTeX у кореневій папці. Нижче приведено невеликий гайд по установці пакету шрифтів psyr, який на мою думку є обов’язковим для верстки документів саме у країнах СНГ, ДЕСТ яких дуже відрізняється від західних країн. Для налаштування шрифтів, потрібно спочатку завантажити дистрибутиви пакету, вони будуть приложені до папки з проектом по створенню шаблону БКР у LaTeX. Далі вміст дистрибутивів потрібно розпакувати у кореневу папку програмного забезпечення , яка знаходиться за адресом C:\Program Files\MiKTeX 2.9 для x-64 бітних систем, та C:\Program Files(x86)\MiKTeX 2.9 для x - 32 бітних. Потім користувачу потрібно натиснути комбінацію

клавіш Win + R , та вписати у строку команду initexmf --edit-config-file dvips, після чого буде відкрито текстовий документ у якому потрібно зберегти такий текст - p +pscyr.map (рис.11).

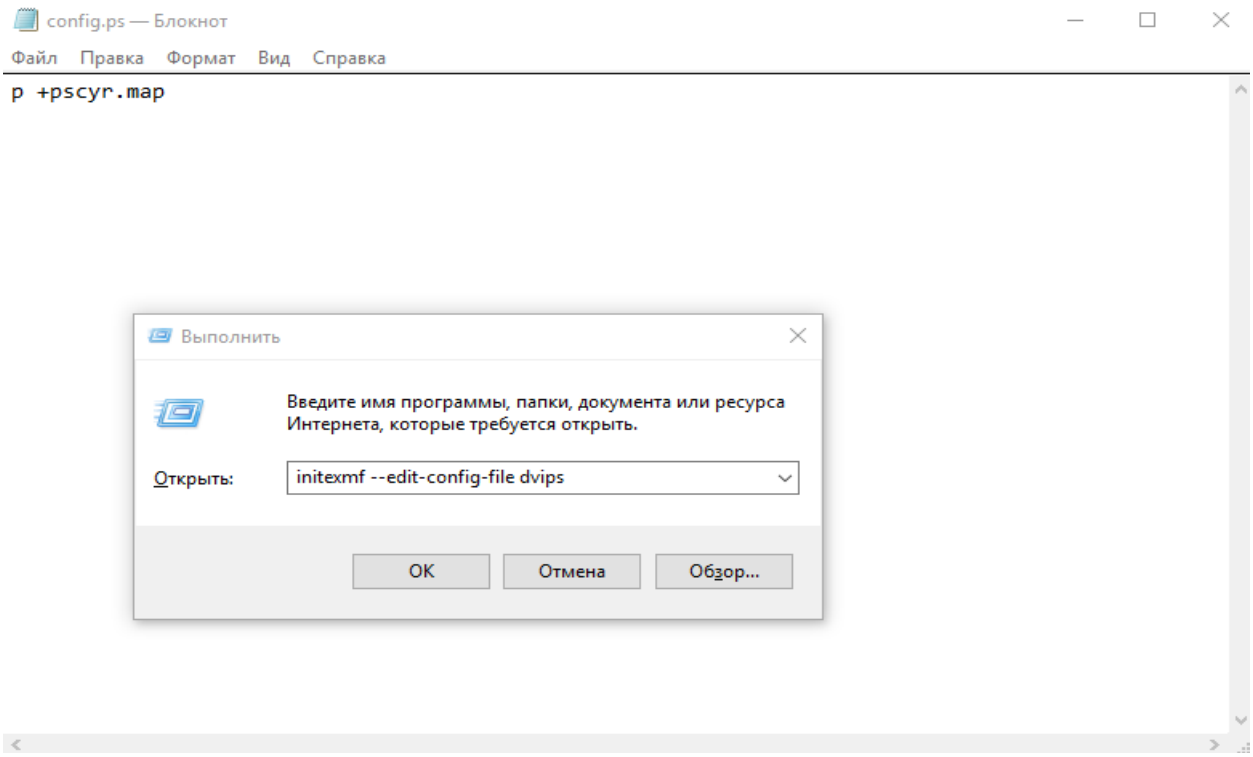


Рисунок 11 – Приклад виконання послідовності налаштування

Це перший крок у налаштуванні шрифтів, саме тому правильність вводу команди у блокнот є дуже важливою. Одразу після цих дій, буде проведено зміни в реєстрі Windows, які стосуються нашого програмного забезпечення. Наступним кроком повторюються ті ж самі дії, однак вже вписується команда initexmf --edit-config-file pdftex, а у блокноті тепер треба зберегти таке речення +pscyr.map [16] [1)]. Після цього користувачу потрібно так само по аналогії з першим та другим разом, натиснути комбінацію клавіш, та ввести наступну команду initexmf --edit-config-file updmap, вона

[1)] [16] Установка PSCyr для Latex (загол. з екрану). URL: http://blog.harrix.org/article/444 (дата звернення: 23.04.2020).

відкриє документ updmap.cfg, у який потрібно вписати наступний набір символів - Map pscyr.map (рис.12).

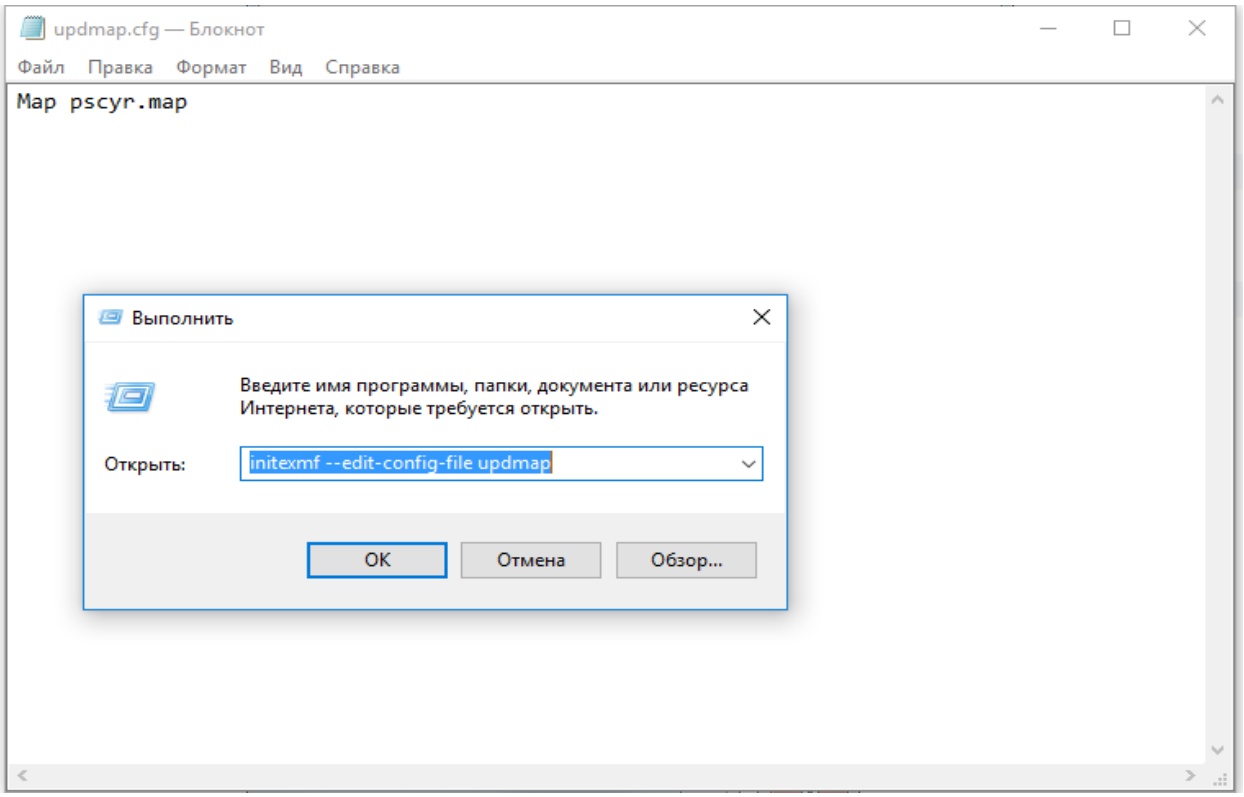


Рисунок 12 – Приклад виконання послідовності налаштування

Далі записуємо у строку виклику елементів команду update. Після цього відкриється командний рядок на декілька секунд, та відбудеться оновлення, а потім він автоматично закриється. Ще раз потрібно відкрити документ за допомогою строки виконання елементів системи (Win+R). Команда, яка дасть нам доступ до файлу конфігурацій, має назву initexmf --edit-config-file dvipdfm. У файлі потрібно зберегти наступну інформацію - f pscyr2.map. Потім вписуємо initexmf -u, та чекаємо хвилину. Після того, як командний рядок автоматично закриється, нам знадобиться оновити базу даних пакетів у програмі MikeTeX. Оновлення БД у даному програмному забезпечені робиться таким, чином: спочатку користувачу треба перейти у режим адміністрації, потім натиснути на вкладку Task, яка знаходиться на

горі інтерфейсу MikeTeX. У випадаючому вікні буде команда Refresh names database (рис.13). Після натиснення на неї одразу вписуємо у строку виконання останню команду mkfntmap. Тепер шрифти підключені із набором тексту не виникне питань.

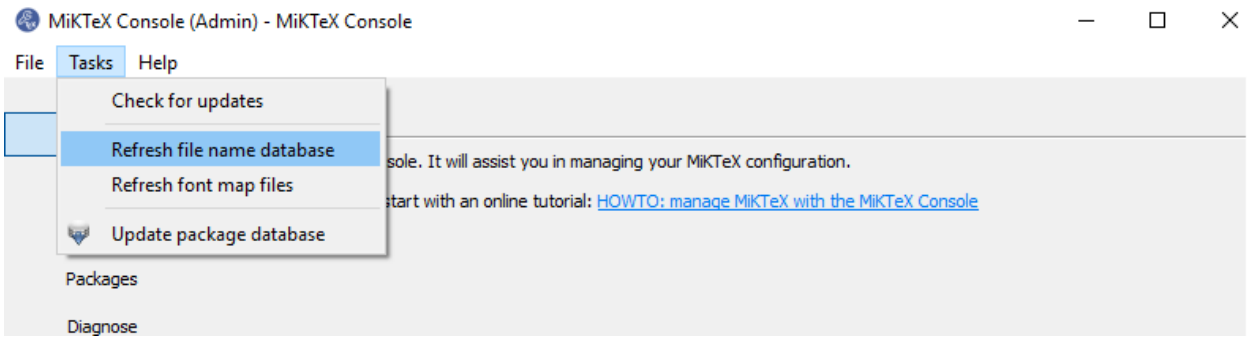


Рисунок 13 – Процес оновлення БД MikeTeX

Для системи Linux, завантаження додаткових програмних забезпечень відбувається наступним чином. Якщо користувач використовує версію системи Debian чи Ubuntu, для того щоб скачати TeXLive (являється ПЗ для загрузки пакетів в системі Linux), потрібно в терміналі прописати таку команду - apt-get install texlive. Після установки TeXLive, треба довантажити за допомогою команди - yum install tetex, ще деякі дистрибутиви, які зроблять процес верстки більш якісним.

Далі, що б почати роботу, потрібно відкрити редактор. Система має дуже добрий, вбудований редактор Emacs (рис.14). Після налаштування документу, файл потрібно запускати у форматі latex !tex.tex, де !tex – ім’я фалу.

Завдяки цим командам, користувач явно звертається до системи LaTeX, перебуваючи у терміналі ОС Linux.

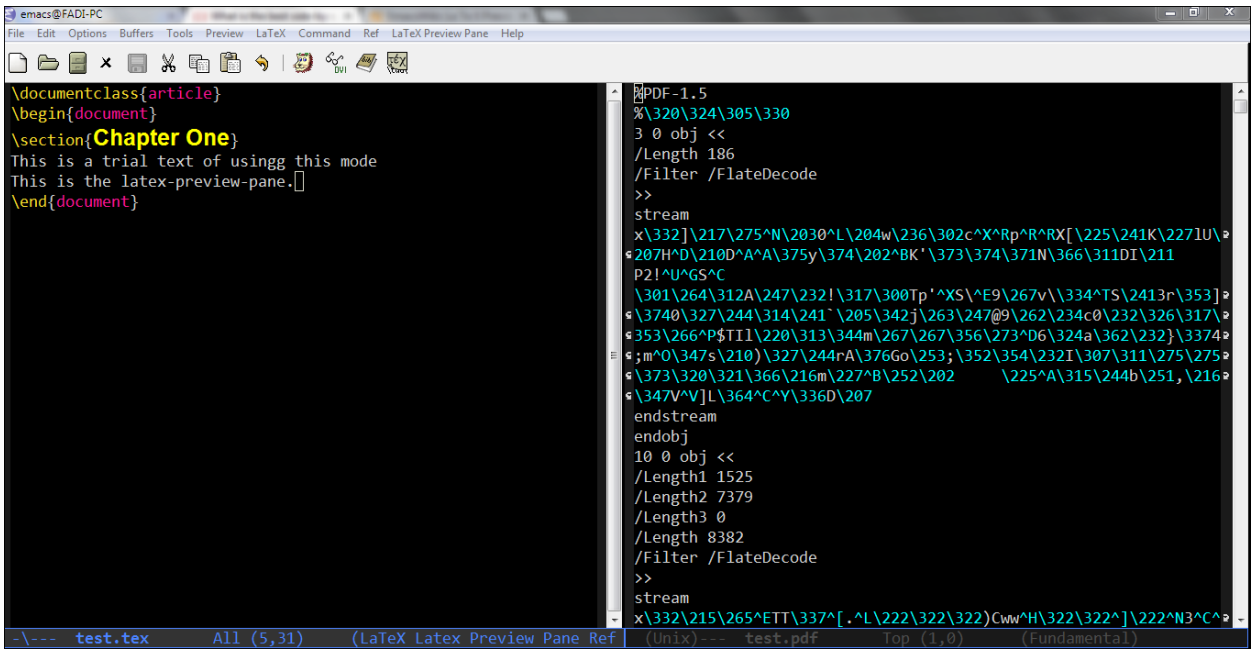


Рисунок 14 – Інтерфейс редактора Emacs

Якщо порівнювати ефективність роботи з системою на ОС Windows і Unix, то можна зробити висновок, що працювати з LaTeX на Unix подібних системах набагато легше, через велику кількість вбудованих редакторів, а також якісну адаптивність саме TeX/LaTeX під них. Однак на сьогоднішній день на території країн СНД процент користувачів ОС Windows значно перевішує тих людей, які користуються іншими системами. Саме тому у даній БКР, є детальний опис по налаштуванню системи LaTeX під ОС Windows. Вище приведено опис шляху для установки та налаштування додаткового програмного забезпечення для роботи саме з TeX/LaTeX.

2 СТРУКТУРА СИСТЕМИ TEX/LATEX

Структура LaTeX є дуже неоднорідною та складною. Однак завдяки цьому система має так багато можливостей по роботі з документом. Чим більше до системи підключено пакетів, тим більше у користувача в запасі команд, для верстки свого проекту.

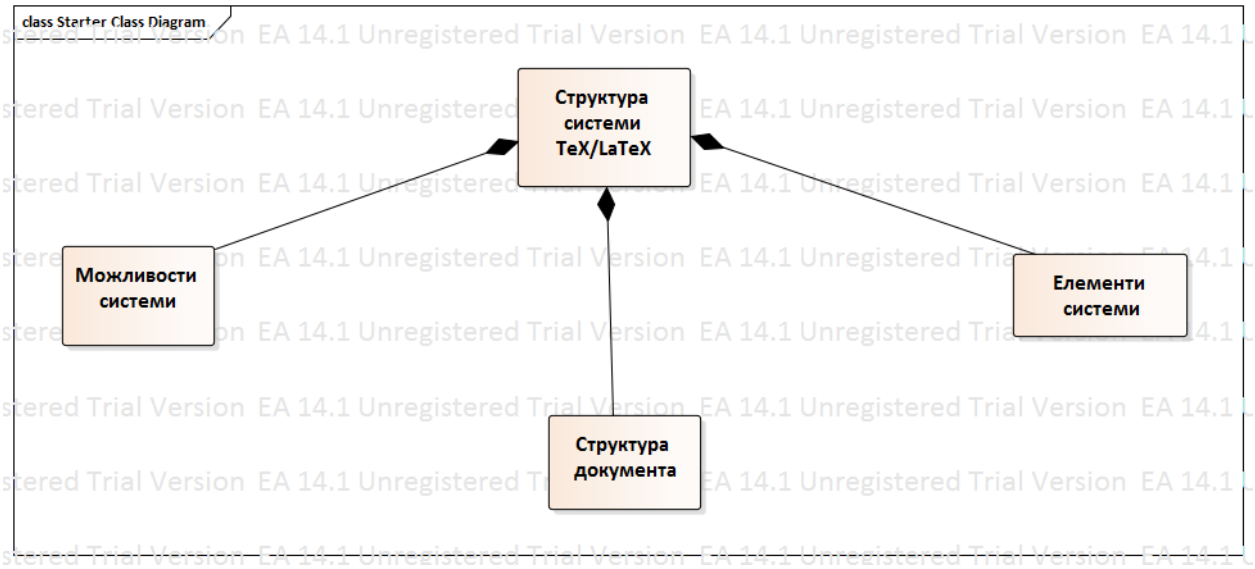


Рисунок 15 – UML діаграма структури LaTeX

Як можна бачити, на (рис.15) зображено три макрорівня, на яких будується вся система. Далі буде детальніше описано структури частин цієї діаграми, а зараз потрібно розглянути кожну ділянку окремо, для найкращого розуміння їх зв’язку з системою LaTeX. Перший рівень відповідає за можливості системи. Слід призначити, що розробники надали користувачам забагато привілеїв. Другий рівень дає уявлення про структуру документів, які створюються у LaTeX в цілому, та описує можливості кожного структурного елементу у документі. Та нарешті третій рівень, який містить безліч різноманітних пакетів, та команд. Цей набор інструментів, дозволяє user’у створювати документацію будь – якого рівня. Завдяки відкритості коду системи, та її програмного забезпечення, структура TeX/LaTeX постійно

ускладнюється самими користувачами, які мають можливість впроваджувати свої власні ділянки коду, та тим самим оновлювати систему [17] [1)].

2.1 Можливості системи TeX/LaTeX

LaTeX має цілу купу можливостей, починаючи від розмітки тексту і закінчуючи особливим підходом до оформлення формул. На (рис.16) приведена UML діаграма, яка дає розуміння про те, з яких саме частин складається блок можливостей системи.

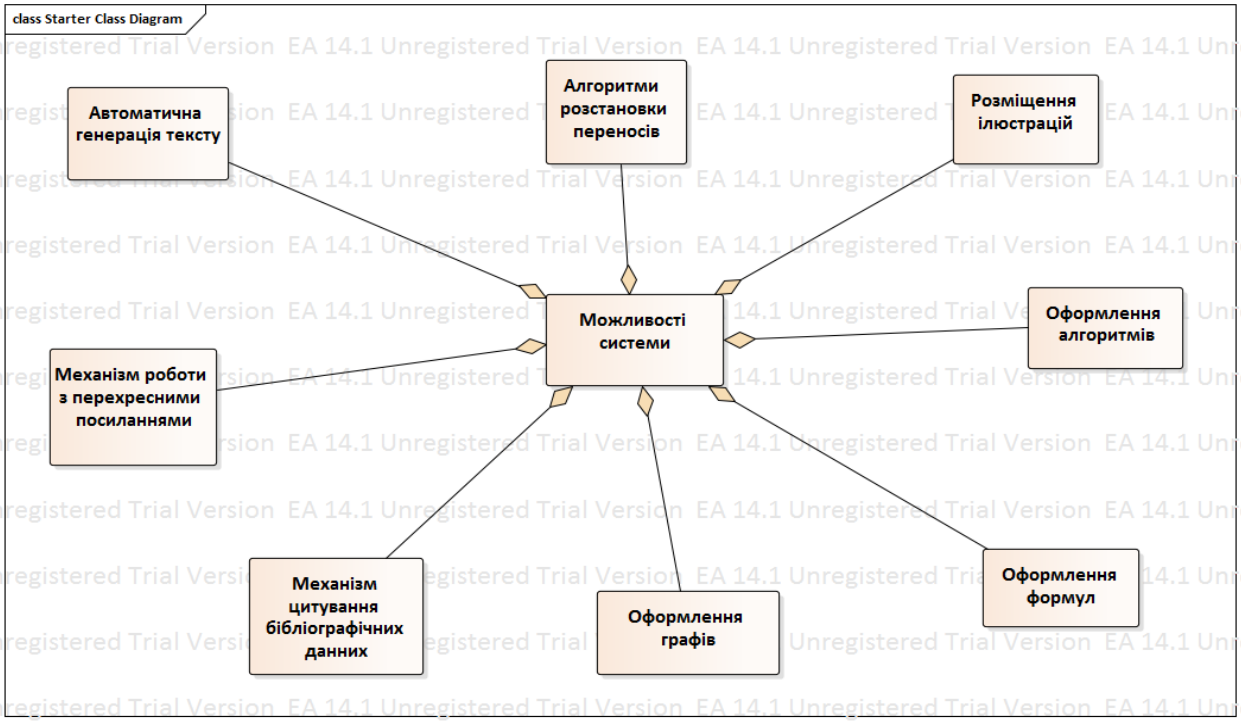


Рисунок 16 – UML діаграма можливостей LaTeX

Розглядаючи рисунок ми можемо бачити, що UML діаграма включає такі компоненти, як: алгоритм розстановки переносів, розміщення ілюстрацій, оформлення алгоритмів, оформлення формул, оформлення графів, механізм цитування бібліографічних джерел.

[1)] [17] Степанова Е.Н. Издательская система LaTeX (навч.посібник). Петрозаводск: Издательство Петрозаводского университета, 1999. 155с.

Також механізм робот з перехресними посиланнями, та автоматична генерація змісту. Розглянемо кожен елемент окремо. Почнемо з алгоритму розстановки переносів. Цей алгоритм дає системі можливість не тільки правильно переносити слова у тексті на новий рядок, а ще й дозволяє користувачу визначати ширину пробілу між словами, та балансувати текст у абзацах [18] [1)].

Далі нами буде розглянуто можливість системи розміщувати у тексті такі плаваючи елементи, як таблиці та ілюстрації. Завдяки спеціальним командам, користувач має змогу створити власну таблицю, яка буде впроваджувати всю важливу інформацію. Також у LaTeX є можливість, спеціальним кодом малювати прості та складні геометричні елементи [19] [2)]. Що стосується звичайної вставки елементів, то тут все дуже просто. Користувачу треба закинути картинку, яка повинна мати одне з стандартних розширень прийнятих для зображень, у папку проекту. Після чого він зможе відобразити цей елемент у себе в документі, за допомогою однієї строки коду. LaTeX надає своїм user'ам можливість підписувати плаваючі документи, та створювати на них посилання. Всі ці команди будуть детально розглянуті у пункті 3.2 бакалаврської кваліфікаційної роботи. Ще система включає в себе, таку цікаву можливість, як оформлення алгоритмів. Ця перевага дає можливість оформлювати алгоритми і частини тексту, які були взяті напряму з файлів, та підкреслювати їх. Також слід розглянути особливе ставлення системи до створення формул, про яке неявно вже згадувалось у попередньому параграфі даної БКР. У LaTeX існує безліч інструментів по роботі з формулами, тому що ця система науково орієнтована, та користується великою популярністю у математиків та фізиків [20] [3)]. У цієї системи є також механізм цитування бібліографічних посилань, тобто

[1)] [18] Львовский С. М. Набор и верстка в пакете LATEX (навч.посібник). Москва: Космоинформ, 1995. 374 с.

[2)] [19] Гуссенс М. Путеводитель по пакету LaTeX и его графическим розши-рениям (навч. посібник). Москва: Мир, 1999. 606 с.

[3)] [20] Богоявленская О.Ю. LaTeX для математиков (навч. посібник). Петрозаводск: ПетрГУ, 2016. 26с.

користувач може процитувати елемент бібліографії, прямо у середині тексту за допомоги спеціальних рядків коду [21] [1)]. Дуже важливою перевагою є те, що LaTeX надає змогу користувачам будувати та оформлювати графи, схеми, діаграми, тому що ці компоненти дуже є важливою частиною абсолютно будь – якої технічно – наукової роботи.

Не менш важливою є можливість створення перехресних посилань та роботи з ними. Саме ці посилання дають можливість користувачу, чітко розмічати та позначати, до якого посилання відноситься та чи інша частина тексту, або формула. Та нарешті останній елемент, який маж назву автоматична генерація змісту. Його суть полягає у вмінні LaTeX налаштувати автоматичне створення об’єктів шляхом додавання деяких команд у преамбулу документа [22] [2)]. Завдяки цієї можливості, рядовий користувач зможе задати налаштування для будь, якого елементу роботи, завдяки декількох рядків програмного коду на початку документа. Наприклад є можливість оформити згідно з ДЕСТ бібліографічний список, чи оголошення. Всі ці властивості системи, дуже добре допомагають користувачу у оформленні свого проекту чи наукової роботи.

Слід також зазначити, що велика кількість можливостей LaTeX обумовлена, тим, що система є орієнтованою на в першу чергу на зручність роботи для звичайного user’a і не несе в собі комерційного підґрунтя. Якщо ж розглянути таке програмне забезпечення, як MS Word, яке має платну ліцензію, то стає очевидною перевага LaTeX в цьому факторі. Саме тому, країни заходу вже багато років, віддають перевагу саме системі LaTeX. На їх думку, ця система є геніальним рішенням усіх проблем пов’язаних з створенням документу, або його редагуванням. Але найважливіший фактор, який приніс LaTeX’у таку шалену популярність, це його гнучкість, та

[1)] [21] Douglas Downing and Michael A. Covington. Dictionary of Computer and In- ternet Terms. 8th ed. Barron’s Educational Series. New York: Woodbury, 2003. 592p.

[2)] [22] The beamer class. User Guide for versio 3.23. (загол. з екрана). URL: http://bitbucket.org/rivanvx/beamer (дата звернення 20.11.2019).

автоматизованість. Також система дозволяє своїм користувачам, дуже легко змінювати її стандартне оформлення, на таке, яке потрібно саме користувачу.

2.2 Структура документу в системі TeX/LaTeX

LaTeX документ можна поділити всього на дві частини: преамбулу та тіло документу (рис.17).

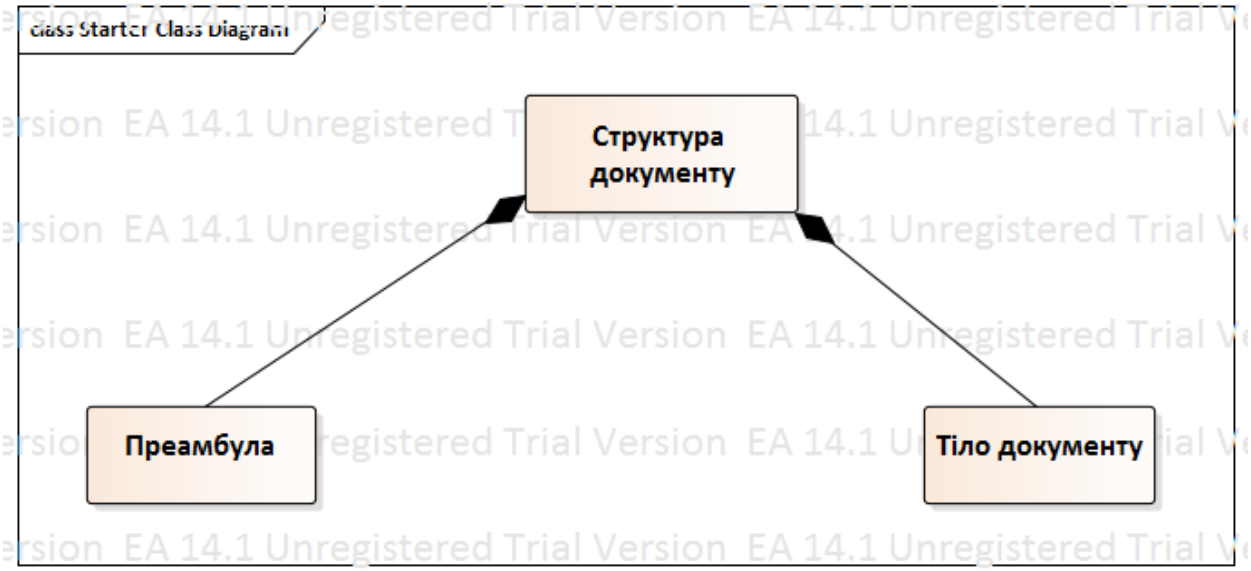


Рисунок 17 – UML діаграма структури LaTeX документу

Ці компоненти служать для тонкого налаштування документів будь якого типу. Преамбула є по суті файлом у якому йде налаштування усіх частин проекту. Користувач у преамбулі може задавати структуру, як і всього документу одночасно, так і його компонентів окремо [23] [1)]. Налаштування здійснюються в основному за допомогою команд \usepackage{}, які підключаються до проекту пакети, назва яких подана у фігурних дужках, а також власні команди користувача, які були створенні саме їм, за допомогою командного оточення \newcommand{}.

[1)] [23] Guidelines for the use of Beamer (загол. з екрана).URL: http://www1.uwindsor.ca/math/sites/uwindsor.ca.math/files/08-02.pdf. (дата звернення 20.11.2019).

Власні команди також використовуються зазвичай для тонкого налаштування деяких елементів, які не подаються налаштуванню макрорівня. По статистиці, правильно заповнити преамбулу проекту, це вже 80 % успіху. Саме після введення останньої команди преамбули, користувач повинен перший раз провести компілювання документу, щоб виявити помилки, або ж впевнитись у правильності набраних команд. Якщо, преамбула проекту була зібрано правильно, користувач може приступати до написання самого тіла тексту. Приклад преамбули наведено на (рис.18). Слід зазначити, що у випадку коли користувач заповнив преамбулу не повністю і перейшов до оформлення тексту, немає нічого жахливого, тому що система за рахунок своєї гнучкості, влаштована так, що user'у за бажанням, надається змога редагувати код преамбули в будь-який проміжок часу.

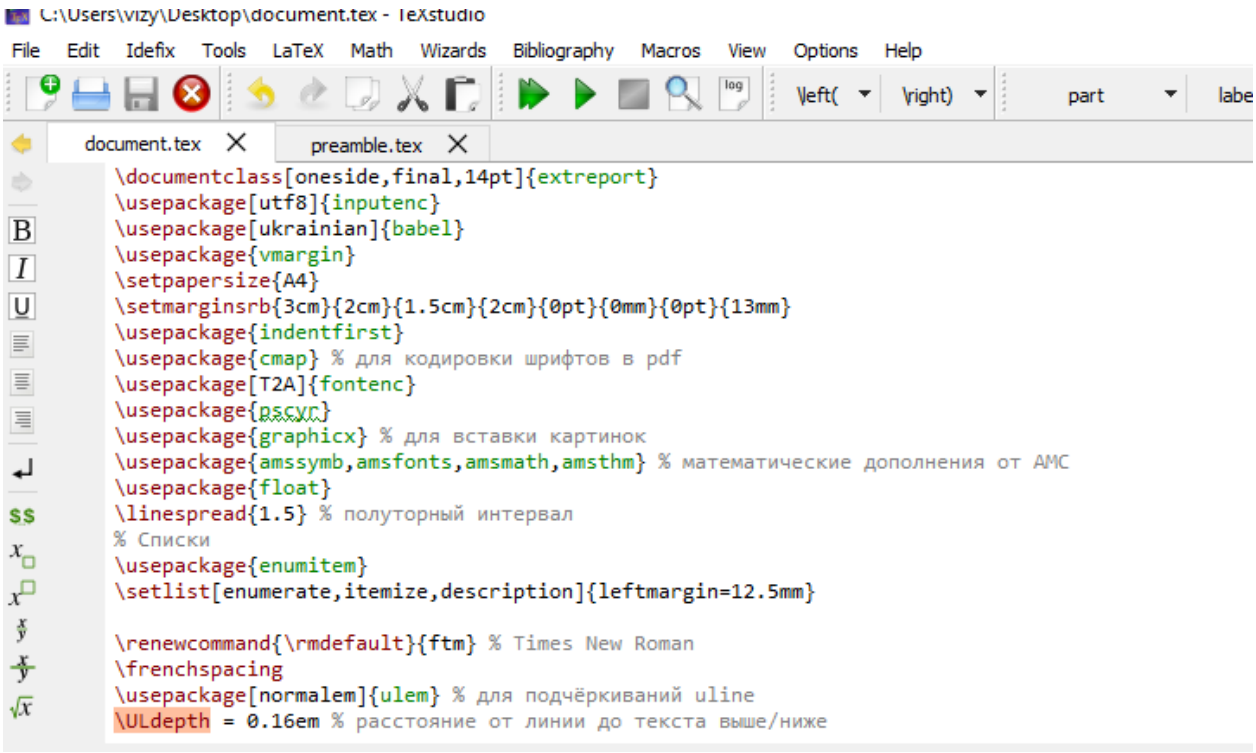


Рисунок 18 – Преамбула документу LaTeX

Після преамбули зазвичай починає оформлюватись тіло документу. Зробити його не так важко, як преамбулу, але воно потребує у рази більше

часу. Це обумовлено тим фактом, що тіло документу містить набагато більше текстової інформації в цілому, а також командного коду. Для того щоб добавляти важливі елементи роботи, для кожного з них потрібно прописувати спеціальний набір команд, будь то команди для створення таблиці, або ж команди для збірки бібліографічного списку. Оточення тіла проекту починається з команди \begin{document}, а закінчується \end{document} [24] [1)].

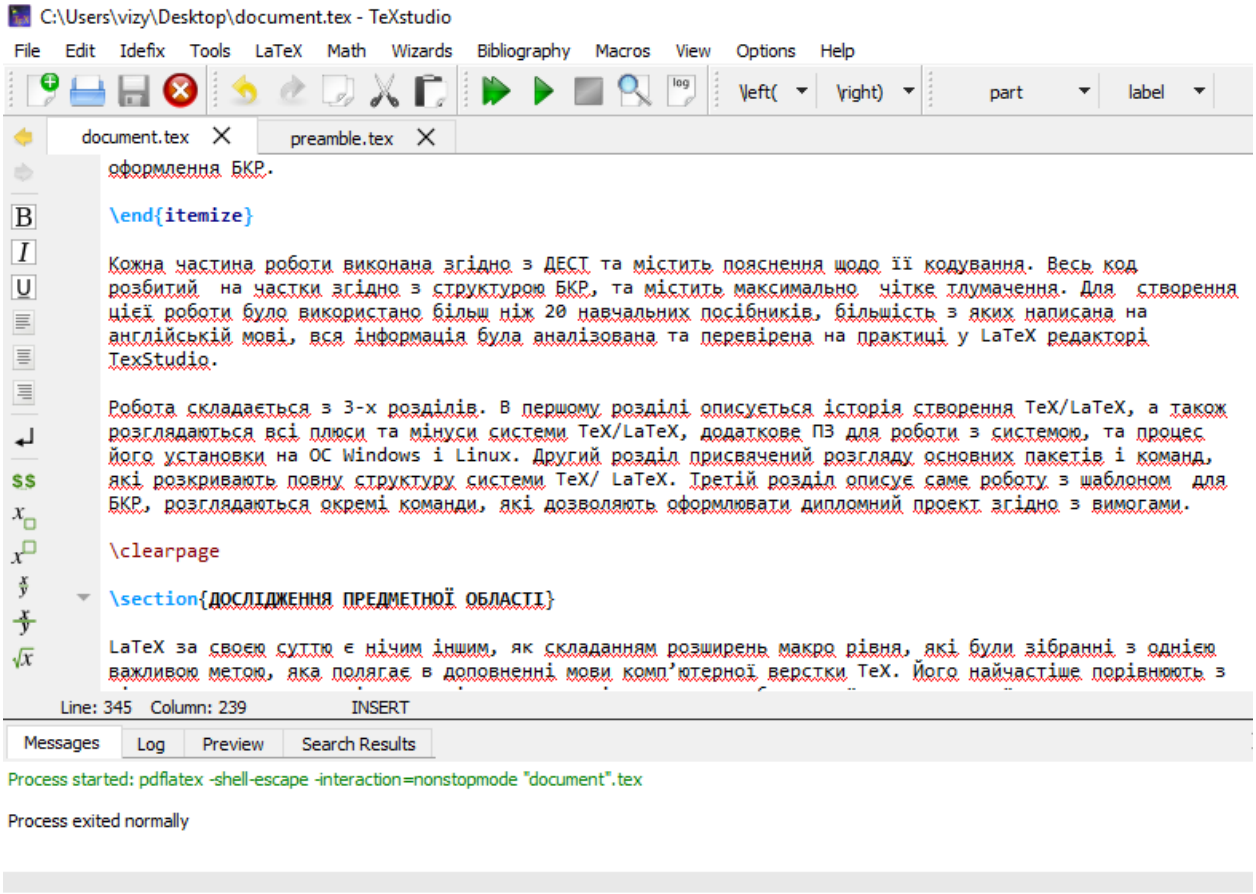


Рисунок 19 – Тіло документу LaTeX

Вся інформація, яка буде знаходитись після команди end, не буде відкомпільована, та представлена у явному вигляді. Це відбувається завдяки

[1)] [24] Lamport L. Latex: A Document Preparation System (навч.посібник). 2 nd. ed. London: Addyson – Wesley,1994. 272p.

тонкого налаштування системи, її розробниками. Вони задали оточенню тіла документа спеціальні теги, схожі на HTML код, та зробили рівні у яких текст не відображається, для того щоб полегшити роботу компілятору, та заощадити час користувача при пошуку помилок. Текст написаний в тілі документа також можна виділяти у оточення, для того щоб наприклад змінювати шрифт, та жирність тексту. Актуальність цього процесу полягає у тому, що користувачу може бути необхідно якось змінити стиль частини тексту, при цьому залишивши деяку частину без змін. Це можна назвати, ще блочним поділом тексту. На (рис.19) представлена маленька частина коду тіла даної БКР.

2.3 Структурні елементи системи

Описати структурні компоненти системи TeX/LaTeX можливо завдяки всього двом елементам. Вони мають назву пакети і команди. LaTeX містить багато команд, які були передбачені його розробниками, як стандартні інструменти для верстки документів у системі. Такі команди носять в основному управлінський характер, та служать для того, щоб підключати пакети, та робити тонке налаштування проекту [25] [1)].

[1)] [25] Knyt D.TeX for beginner (навч.посібник). 1 st. ed. New-Yourk: Williams,1993. 560с.

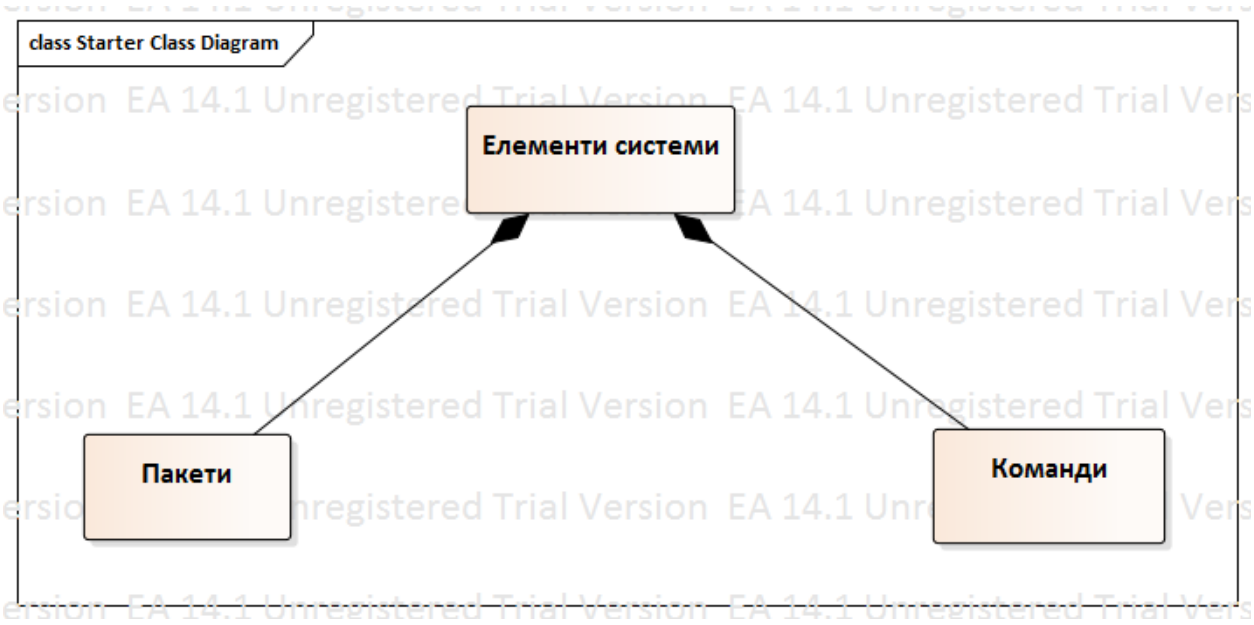


Рисунок 20 – Елементи системи LaTeX

Також у системі є можливість підключення інших команд. Вони проваджуються у проект разом з пакетами. На сьогоднішній час існують пакети на будь-який випадок. Саме це дозволяє користувачу самому вибирати, які саме дистрибутиви йому підключати для налаштування свого документу. Як вже було згадано раніше, пакети підключаються за допомогою спеціального програмного забезпечення. Є можливість закачування пакетів з Інтернету, у випадку коли елементи пакету не можливо знайти у дистрибутивах спеціального ПЗ [21] [1]. На (рис.20) зображена структура елементів системи.

Користувач після установки MikeTeX може одразу, ще до роботи, скачати не тільки обов'язкові пакети, які впроваджує сама система, але й будь які пакети на вибір з даного списку (рис.21). Це дуже цікаве рішення тому, що завдяки цьому сховищу, пакети можуть впроваджуватись у систему, якщо у преамбулі документу є команда, яка звертається до них. Саме цей функціонал додає системі додаткову гнучкість.

[1] [26] И. А. Котельников, П. З. Чеботаев. Издательская система LATEX 2ε (навч. посібник). Новосибирск: Сибирский хронограф, 1998. 496с.

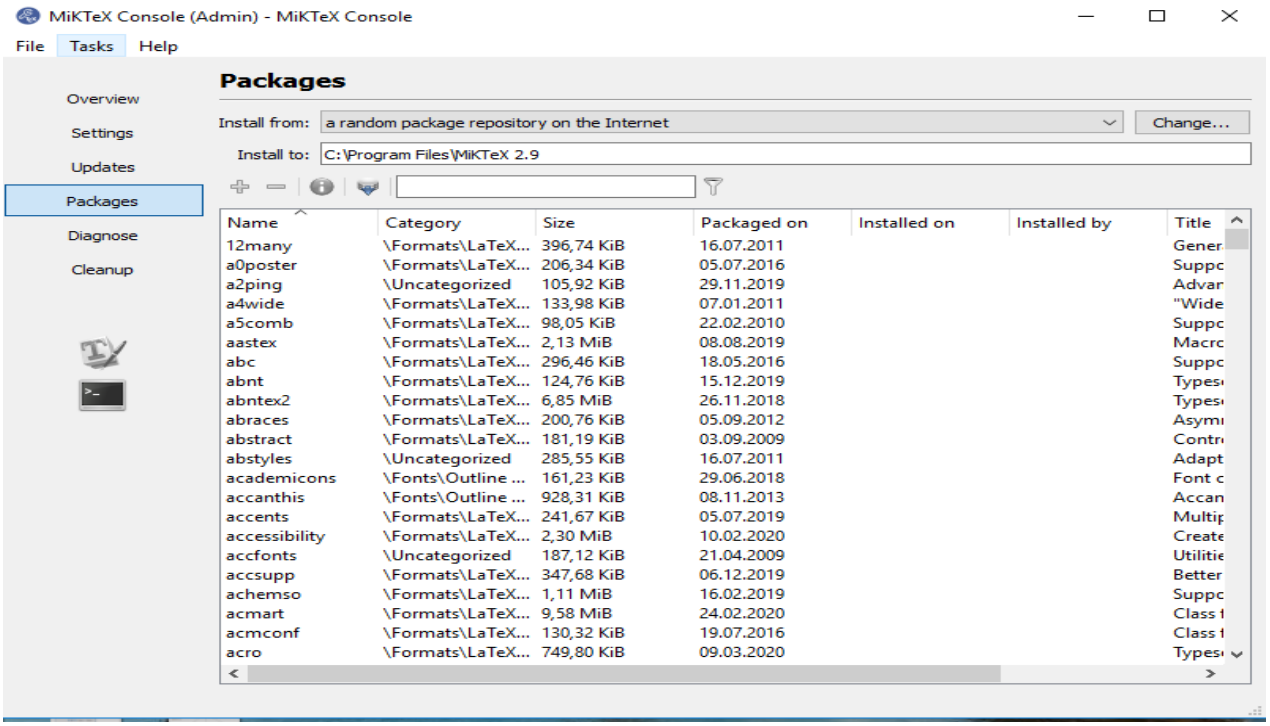


Рисунок 21– Сховище пакетів LaTeX

3 РОЗБІР ШАБЛОНУ БКР

У цьому розділі буде розібрано структуру шаблону БКР, а також описано всі команди, які використовувались для досягнення поставленої мети.

3.1 Структура шаблону БКР

Для більш детального розуміння структури даної БКР, розглянемо UML – діаграму (рис.22).

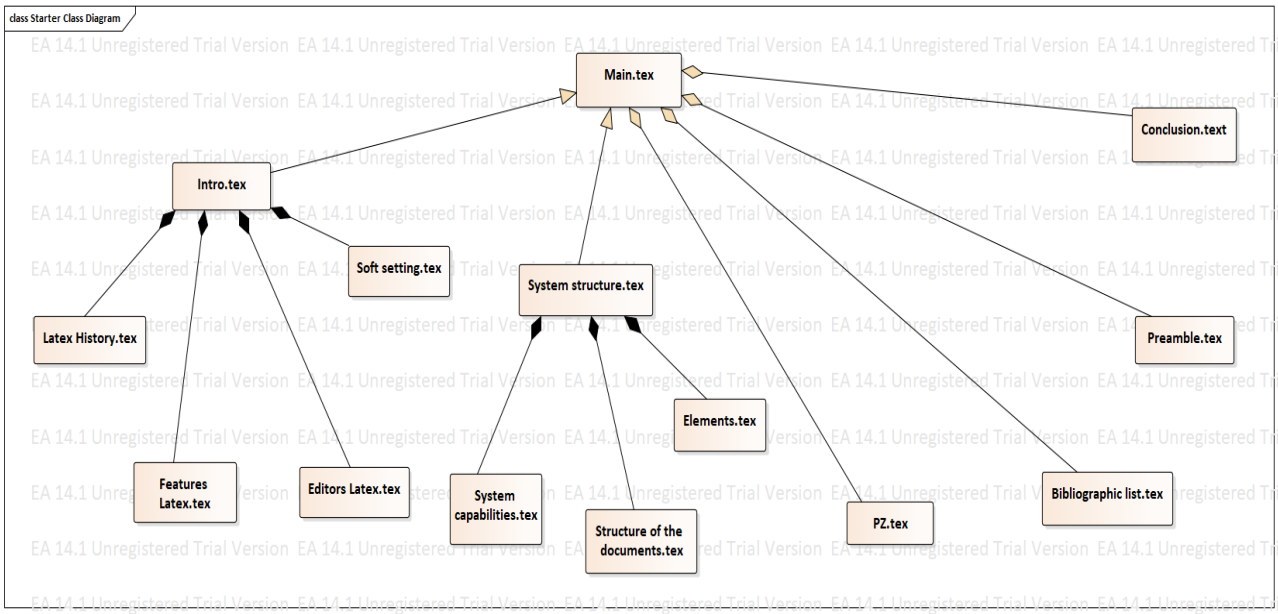


Рисунок 22 – Структура БКР у UML диаграммі

Як можна бачити з UML – діаграми, головним файлом у якому зібрані всі компоненти БКР являється Main. Він включає в себе Intro, який впроваджує введення, яке поділяється ще на чотири файли: LaTeX History, Features LaTeX, Editor LaTeX, Soft setting, які відповідають за історію, особливості системи, а також дають інформацію про його програмне забезпечення. Далі підключається файл System structure, який впроваджує додаткові файли: elements, structure of the documents, system capabilities. Вони відповідають за функції у БКР відповідно своїм назвам. Також підключається преамбула, яка містить усі налаштування документу, та PZ , який включає завдання до бакалаврської дипломної роботи. Також підключається бібліографічний список і висновки.

3.2 Команди, які використовувались при створенні БКР

Команди у системі LaTeX є дуже важливим інструментом для роботи з документами. Саме завдяки ним системні проекти створюються та оформлюються. Для наочності, розглянемо деякі команди, які були записані

в преамбулі документу, та являються грубо кажучи його налаштуваннями. В преамбулі зазвичай вказується вся інформація про документ, наприклад: параметри сторінки, абзацний відступ, міжрядковий інтервал, розмір шрифту, мова написання, сам шрифт та ін. Навіть цього вже достатньо, щоб зверстати, який небудь не складний реферат, але для більш складних наукових робіт, використовується набагато більше команд, всі вони будуть розглянути нижче, та розібрані досконало, з приведенням прикладів.

Почнемо з самих основних команд, без яких неможливо написати навіть дуже простий текст та скомпілювати його. Перша команда є елементарною, однак її використання вирішує багато питань.

Це команда \documentclass{}, в даній БКР вона була використана мною таким чином: \documentclass [oneside, final, 14pt]{extreport}, розглянемо більш детально. Завдяки її присутності у коді, користувач має можливість вибрати з яким типом документа він хоче працювати, виходячи з його вибору LaTeX буде надавати нам набір команд з якими можна працювати або тільки виключно у цьому класі документів, або у всіх класах. Існує багато різноманітних класів, таких як: article – клас який дає системі розуміння, що користувачу потрібно розробити саме статтю, extrarticle – розширений клас від батьківського класу article, має декілька додаткових команд які розширюють його, та book – клас для користувачів які пишуть книгу, їм користуються в основному письменники, report та extrareport – клас для написання різних типів рапортів, та ін [27] [1]. У своїй роботі я використовував саме клас extrareport, тому що він більш за інших підходить мені для структурування саме мого диплому. Як видно клас задається у фігурних дужках, вони є основними у системі, я би позначив їх як перш пріоритетні дужки. Також є необов’язкові дужки, які позначаються двома символами квадратних дужок, в них зазвичай заносять які небудь параметри, в моєму випадку було завдано розмір шрифту, також пояснення що документ

---

[1] [27] П.Каров. Шрифтовые технологии. Описание и инструментарий. Москва: Мир, 2001. 454с.

находиться в кінцевому стані і що він не є чернеткою, цього ефекту було досягнуто за допомогою параметру final, якщо користувачу треба щоб система вважала його документ чернеткою, він може змінити значення на draft. І на решті останній параметр oneside, позначає що наш документ буде друкуватись по одній сторінці на лицьовій частині листа, так потребує новий ДСТУ. Можливо зробити по іншому, на приклад завдавши значення double-side [28] [1)], на кожному листі, буде друкуватися по дві сторінки документу, с фронтальної та тильної сторони.

Наступні дві команди є також невід'ємною частиною будь якого проекту створеного у LaTeX. Будемо розглядати їх по черзі. Слід запам'ятати, що 90% команд, які підключаються до преамбули документа, починаються так: \userpackage. Це ж саме і відноситься до цих двох команд. Далі слід розібрати їх досконаліше. У коді вони будуть позначені таким чином:

```
\usepackage[utf8]{inputenc}
\usepackage[ukrainian]{babel}
```

Перша несе відповідальність за кодування документу, а друга за його мову. Ключові слова слід розглядати у фігурних дужках цих команд. Це inputenc – для кодування елементів проекту, а також babel – для вибірки потрібної нам мови. Також не треба забувати про параметри у квадратних дужках. Для кодування, мною було підібрано значення utf8, оскільки моя версія операційної системи підтримує переважно цей тип, однак на більш ранніх версіях Windows рекомендується користуватися таким типом кодування, як cp1251 [29] [2)]. Для ОС Linux існує спеціалізований тип кодування, який є дуже популярним у користувачів, це koi8-r.

[1)] [28] D. E. Knuth. The TEXbook, часть A серии Computers and Typesetting (навч. посібник). New-Yourk :Addison – Wesley, 1984. 496p.

[2)] [29] М.Спивак. Восхитительный TEX: руководство по комфортному изготовлению научных публикаций в пакете AMS-TEX (навч. посібник). – Москва: Мир, 1993. 282с.

Далі на черзі в преамбулі моєї роботи, йдуть команди, без яких оформлення документу у системі LaTeX по ДЕСТ було б не можливе. Для того, щоб написати ці команди у термінал, потрібно спочатку підключити спеціальний пакет \usepackage{vmargin}.

Далі на черзі йде саме \setpapersize{A4}. Як ми бачимо, за допомогою цієї команди задається формат сторінки. У своїй БКР я використовував формат А4, тому, що цього потребують вимоги до складання та оформлення дипломної роботи на 2019 – 2020 рік. Однак рядовий користувач, має повне право використовувати будь - який формат, котрий потребує його проектна робота. Цей фактор також є дуже важливим, тому, що наприклад у тому ж MC Word, для того щоб завдати такі параметри сторінки, людині знадобиться більш часу ніж на введення коду у командний рядок редактору LaTeX.

Слід також приділити увагу команді \setmarginstbr {3cm}{2cm}{1.5cm} {2cm}{0pt}{0mm} {0pt}{13mm}. Вона задає поля сторінки, в даному прикладі 3см - це відступ зліва, 2см, 2см - відступи знизу та зверху, та 1.5см – відступ справа. Також заданий параметр у вигляді 13 мм, має на увазі розташування відносно до номеру сторінки [30] [1]. Налаштування цієї команди задано відповідно до вимог оформлення БКР.

Також слід розібрати дуже важливу команду \usepackage {setspace}. Вона підключає пакет, та дозволяє провести налаштування міжрядкового інтервалу саме для виносок, тобто відокремити установки інтервалу основного тексту від тексту виносок [31] [2]. За нею йде пакет \userpackage {indentfirst},

---

[1] [30] Столяров А.В. Сверстай диплом красиво: LATEX за три дня(навч. посібник). Москва: МАКС Пресс, 2010. – 100 с.

[2] [31] Говорухин В., Цибулин Б. Компьютер в математическом исследовании: Maple, MATLAB, LaTeX(навч. посібник). Санкт-Петербург: Питер, 2001. 624 с.

він дозволяє завдати обов'язковий абзацний відступ, після тегов \section{}, \subsection{}, та інших, що мають відношення до структури деякого документа [32] [1)].

Далі будуть розглянуть ще три команди, які також знаходяться на вершині преамбули:

```
\usepackage{cmap}
\usepackage[T2A]{fontenc}
\usepackage{psyr}
```

Пакет cmap надає таблиці зіставлення символів, які роблять PDF – файли, доступними для пошуку і копіювання в засобах їх використання [28] [2)]. Для досягнення правильної генерації документів у LaTeX ви також повинні вибрати кодування шрифту, яка повинна підтримувати певні символи для російської мови, це досягається пакетом fontenc. Останній пакет підключає шрифти, наприклад у моєму дипломному проекті, за допомогою нього, був підключений шрифт Times New Roman.

Дуже популярним інструментом в редакторах типу WYSIWYG, є завдання інтервалу між реченнями тексту. Для того щоб задати цей інтервал у даній системі потрібно ввести у командний рядок таку послідовність букв та символів: \linespread{1.5}. Ця команда була взята з мого проекту, тому, як читач може побачити, заданий полуторний міжрядковий інтервал, повністю відповідає вимогам ДСТУ.

Преамбула, є самою великою частиною у розумінні написання коду користувачем. Там задані не тільки основні параметри для оформлення сторінок, шрифту та завдання мови документу. Можливості коду преамбули також розповсюджуються гіперпосилання, та список літератури, на оголошення та на формування та завдання математичних формул. Ще

[1)] [32] G. Gratzer. First steps in LATEX (навч. посібник). Birkhauser: Springer – Verlag, 1999. 131p.

[2)] [33] Богоявленская О. Ю. Компютернные технологии в математике (загол. з екрану) Петрозаводськ, 2015. URL: http://kappa.cs.karelia.ru/~olbgvl/ktmath/matem-skt.html (дата звернення: 24.09.2019).

можливе підключення до преамбули пакетів, що дозволять user'у завантажувати картинки, та робити під ними записи. Існує безліч команд по завданню нумерації сторінок, команд для поділення тексту документа на частини, та секції, а також такі, що дозволяють користувачу отримувати свої команди шляхом складання тих, що маються в одне ціле. Далі команди будуть приводитись у вигляді боку кода, які відповідають певним функціям , що будуть застосовуватись, як для тексту документу , так і для самого проекту у системі LaTeX. Кожен блок буде взято з мого дипломного проекту, та досконально розібрано, для наочності оформлення преамбулы складного документа.

Також були добавленні установки по роботі зі списками. Вони включають всього дві команди, які мають традиційний синтаксис. Причина їх введення у проект також полягає у стандартах, які диктує новітній ДСТУ. Нижче розглянуто дві команди, які налаштовують списки до потрібного вигляду:

```
\usepackage {enumitem}
\setlist[enumerate,bitemize,description]
{leftmagin = 12.5mm}
```

Перша команда підключає пакет, який дозволяє проводити деякі маніпуляції по налаштуванню потрібного нам елементу. Друга ж команда задає налаштування спискам. Як можна бачити, в квадратних дужках знаходяться необов'язкові елементи. Саме в цьому фрагменті коду, вони служать для вказівки на те, до яких списків буде застосовано зміни. В коді написано, що для нумерованого, ненумерованого, та списку опису, буде застосовано налаштування, яке зрушить список з його елементами на 12.5 мм ліворуч.

Одним з найскладніших етапів налаштування преамбули для БКР, являється підключення пакетів для виносок, та завдання для них установок. Оскільки для виносок LaTeX задає інший шрифт, та інтервал по стандарту,

було прийнято рішення, виправити стандартні регулювання за допомогою такого командного коду:

```
\def\setnotebodyfont{\switchtobodyfont[12pt]
\setupinterlinespace[1.0]}
\userpackage[bottom]{footmisc}
\setlength{\skip\footins}{1cm}
```

Перша команда задає правильні по ДСТУ шрифт та міжрядковий інтервал виносок. Друга виправляє становище виносок у документі, та прижимає їх вниз за допомогою пакету footmisc [34] [1)].

Слід зазначити, що налаштовувати положення виносок можна прижимаючи їх до всіх сторін сторінки. Остання команда задає відступ між текстом та виносками, за для найкращого вигляду вихідного документу. Також корисним інструментом при написанні БКР є підкреслення слів. Для того щоб підключити його до преамбули потрібно ввести декілька команд. Вони також допоможуть у налаштуванні деяких можливостей підкреслення відносно тексту. За стандартом підкреслення у LaTeX вимкнено. Розглянемо ці команди:

```
\usepackage[normalem]{ulem}
\ULdepth = 0.16em
```

Як можна побачити, саме перша команда відповідає за підключення підкреслення у систему, однак в квадратних дужках вказано, ще й аргумент. Він задає нормальну жирність для підкреслення. Друга команда ж задає відступ він тексту до підкреслення.

Важливу роль у коді преамбули має блок команд, який відповідає за нумерацію сторінок, а також за розміщення номера на сторінці, якщо детальніше можна вказати де саме треба розмістити номер, зверху чи знизу, з правого боку чи зліва. Далі буде приведено блок коду з преамбули документу, який зазвичай використовують для якісної нумерації сторінок:

[1)] [34] GitHub. Amet13/bachelor-diploma (загол. з екрана). URL: https://github .com/amet13/bachelor-diploma/inc/peamble.tex. (дата звернення 27.12.2019).

```
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\fancyhead[R]{\textrm{\thepage}}
\fancyheadoffset{0mm}
\fancyfootoffset{0mm}
\setlength{\headheight}{17pt}
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0pt}
\fancypagestyle{plain}{
\fancyhf{}
\rhead{\thepage}
}
```

Починається блок коду саме з команди \usepackage{fancyhdr}, яка підключає пакет колонтитулів, який і дає змогу користувачу завдати нумерацію. На черзі йде така команда, як \pagestyle{fancy} [35] [1)]. Вона відповідає за можливість зміни стилю сторінки, так є дуже важливою. Команда \fancyhf{} виконує очистку поточних значень.

Далі розглянемо наступні команди \fancyhead[R] {\textrm {\thepage}}, \fancyheadoffset {0mm}, \ancyfootoffset {0mm}. Перша команда з перерахованих відповідає конкретно за установку верхнього колонтитулу, далі йдуть до- даткові команди. Вони вносять в налаштування системи, розмір відступу від самого номеру, до полів документу. Команда \setlength{\headheight}{17pt} задає відступ від верхнього поля до номера. Слід відмітити, що лінія розділення між текстом та виносками, по стандарту має довжину, яка не відповідає потребам ДСТУ, тому для виправлення цього явище, було розроблено невеликий скрипт. Сам скрипт наведено нижче, з описом його роботи, та тонкощів користування:

```
\renewcommand{\footnoterule}
{\raggedright\rule{4cm}{0.4pt}
\vspace—{0.6ex}}
```

[1)] [35] Диплом бакалавра в LaTeX в 150 строк (загол. з екрана). URL: https://www.pvsm.ru/diplom/8387/print/. (дата звернення 23.11.2019).

Його суть полягає у тому, що він перевизначає стандартні правила для виносок, тим самим скорочуючи лінію розділення до 40 мм, що є прийнятним для стандартів ДСТУ.

Також требо обов'язково розглянути підписи під рисунками, та формулами, тому що це дуже важлива частина, написання та оформлення будь якої роботи у системі LaTeX.

```
\usepackage[tableposition=top]{caption}
\usepackage{subcaption}
\DeclareCaptionLabelFormat{gostfigure}{Рисунок #2}
\DeclareCaptionLabelFormat{gosttable}{Таблица #2}
\DeclareCaptionLabelSeparator{gost}{~---
~}\captionsetup{labelsep=gost}
\captionsetup[figure]{labelformat=gostfigure}
\captionsetup[table]{labelformat=gosttable}
\renewcommand{\thesubfigure}{\asbuk{subfigure}}
```

Тут за допомогою команди \DeclareCaptionLabelFormat налаштовується новий користувальницький формат оформлення підпису, який потім призначається для всіх малюнків і таблиць документа [36] [1)]. Йога перший параметр #1— це стандартний текст мітки (краще не налаштовувати його окремим рядком, а написати явно, не використовуючи #1), другий параметр #2 — номер малюнка / таблиці. \DeclareCaptionLabelSeparator налаштовує роздільник між міткою підпису (caption label) і безпосередньо її текстом, який задаєте ви. У нашому випадку це середнє тире.

Окремо варто прокоментувати останній рядок. Так як я поділяю іноді ілюстрації на частини, що дозволяє робити пакет subfigure, ці частини потрібно нумерувати. Прийнято нумерувати їх маленькими літерами по порядку - однак в нашому випадку це повинні бути літери кирилиці, а не латиниці! Кириличне уявлення лічильників (за аналогією з відомими уявленнями arabic, roman, latin та ін.) Вже реалізовано в пакеті babel.

[1)] [36] Диплом бакалавра в LaTeX в 150 строк (загол. з екрана). URL: https://www.pvsm.ru/diplom/8387/print/. (дата звернення 23.11.2019).

Називається воно відповідно \asbuk для маленьких кириличних літер, і \Asbuk для великих.

Далі розглянемо такий блок команд, який відповідає за налаштування наочного вигляду для бібліографії. Я використовував його для налаштування інтервалів між посиланнями, для встановлення лічильника сторінок, за допомогою якого, якщо буде добавлена інформація всередину тексту, лічильник почне перерахунок всіх сторінок, і на виході у бібліографічному списку, проставить номера сторінок відповідно до інформації на них.

Також у цьому блоці є команда яка задає відлік з 4 сторінки документу. Нагадаю, що цей блок повинен бути у преамбулі, та підключатись, як інші. Якщо розмова йде про створення бібліографічного списку, а не про його оформлення, то це вже зовсім інша частина проекту, і найчастіше її код розташовується на при кінці листа з командами. Блок для преамбули буде мати такий вигляд:

```
\makeatletter
\renewenvironment{thebibliography}[1]
{\section*{\refname}
\list{\@biblabel{\@arabic\c@enumiv}}
{\settowidth\labelwidth{\@biblabel{#1}}
\leftmargin\labelsep
\itemindent 16.7mm
\@openbib@code
\usecounter{enumiv}
\let\p@enumiv\@empty
\renewcommand\theenumiv{\@arabic\c@enumiv}
 }
 \setlength{\itemsep}{0pt}
 }
 \makeatothe
 \setcounter{page}{4}
```

Також слід зазначити, що в коді також використовуються команди, які задають довжину рядків, а також розмір шрифту, та розташування по лівому краю документу для бібліографічного списку [37] [1)]. Після того, як була

[1)] [37] GitHub. Aoklyunin / CSI_REPORT (загол. з екрана). URL: https://github.com /aoklyunin/CSI_REPORT/blob/master/arhive/Заключение.tex.(дата звернення 24.12.2019).

кратко описана частина коду в преамбулі, можна приступати до опису створення самого бібліографічного списку, для цього нам потребується цілий перечень команд. Зараз читач може побачити фрагмент коду , який був взятий з мого проекту для наочності створення та оформлення бібліографії документу:

```
\begin{thebibliography}{5}
\bibitem{Kopka1999}
Kopka H.,W.Daly.P.A guide to LaTeX:document preparation
for beginners and advanced users. 3rd ed.p.cm. London :
Addison-Wesley, 1999. 609p. (Series Head First).
ISBN  0 -201 -39825 -7
\bibitem{Goosens2004}
Goossens M, Mittelbach F, Braams J., Carlisle D., Row-
ley C.The LaTeX Companion. 2nd ed.p.cm. Boston : Addi-
son-Wesley, 2004. 1119p.
(Series Head First).ISBN 0-201-36299-6.
\bibitem{Kottwiz2011}
Kottwitz S. Latex Beginer's Guide.Olton : Packt Pub-
lishing,2011. 336p.(Series Head First).
ISBN  9781847199867.
\bibitem{Роженко2005}
Роженко А. Искусство верстки в LaTeX'е. Новосибирск:
Изд.ИВМиМГ СО РАН, 2005.398с. ISBN 5-901548-25-6.
\bibitem{Балдин2008}
\end{thebibliography}
\end{document}
```

Починається бібліографічний список з простої команди \begin{the bibliography}{5} [38] [1)]. По суті вона створює список, в який входить кількість елементів, яка задана у фігурних дужках. Далі ми бачимо команди по типу \bibitem{Kopka1999}. У дужках вказано ідентифікатор елементу списку. Ця команда створює рядок який надає інформацію читачу, про те з якого джерела був взятий матеріал у проекті. Вертаючись до ідентифікатору рядку, слід сказати , що він потрібен також, щоб зробити посилання на інформацію з тексту, яка була взята при використовуванні деякого джерела інформації. Щоб зробити посилання на частину тексту, потрібно використовувати таку команду, як \label{Kopka1999}.

[1)] [38] Linux.Org.Ru. Настройка библиографии LaTeX (заг. з екрана). URL: https://www.linux.org.ru/forum/general/11477368 (дата звернення 27.12.2019).

Виходячи із цього ми можемо зробити перехресне посилання на частину тексту, та позначити її номером у квадратних дужках, який відповідає порядковому номеру у бібліографічному списку. Для того щоб, надати системі інформацію, що список закінчено, потрібно написати таку команду, в командний рядок користувача: \end{thebibliography}.

Слід також зробити спеціальні установки в тексті, які дозволять користувачу, легко розібратись з абзацними відступами, а також з переносом слів. Нижче приведено фрагмент коду, який складається з п'яти команд:

```
\parindent 1.25cm
\sloppy
\hyphenpenalty = 1000
\clubpenalty = 10000
\widowpenalty = 10000
```

Перша команда являться налаштуванням саме для абзацного відступу. Вона задає відступ для червоної строки у 1.25 см [39] [1)]. Це є прийнятним стандартом для оформлення документів. Далі йде така команда, яка допомагає документу позбавлятися від переповнення тексту.

Останні три команди доповнюють попередню в повній мірі. Вони додатково, у порядку черги задають документу такі властивості, як частота переносів, заборона розриву сторінки після першої строки абзацу, та заборона розриву сторінки після останньої строки абзацу. Слід також ознайомитись з порядком створення та оформлення оголошення, а також з командами, які потрібні для того, щоб ділити текст документу по параграфам, секціям, та розділам. Існує приблизно сім рівнів, для поділу тексту, але на практиці зазвичай достатньо використовувати три рівня, цього достатньо для створення документів любої складності. Слід приймати до уваги той факт, що для різних класів документів у системі, існують різні структури поділення тексту.

Самим першим ступенем в ієрархій поділення тексту є команда \chapter{Глава}. Вона є дуже важливою, тому що надає можливість user'у

[1)] [39] GitHub. Vrovachev/diplom (загол. з екрана). URL: https://github.com/ vrovachev/diplom/blob/master/gost.tex.(дата звернення 25.12.2019).

ділити документ на глави, для того щоб запобігти змішанню тексту. Далі на черзі команда типу \section{Секція}, яка ділить глави на секції, а також команда \subsection{Підсекція}, котра по аналогії ділить секції, на підсекції. Також читачу буде корисно познайомитись з такими командами, як \paragraf{Параграф}, та \subparagraf{Підпараграф}, зазвичай користувачі використовують максимум 3 рівня, це обумовлено тим, що система LaTeX, має вбудований лічильник, який займається нумерацією рівнів розподілення тексту, і якщо користувач буде використовувати команди 4-го, та 5-го рівнів розподілення, то на виході він отримає такий результат нумерації, як 1.1.1.1.1, це виглядає досить громіздким, як що допустимо, що заданий шрифт це Time New Roman, 14pt. Однак, якщо user навіть змінить тип шрифту, та його розмір, результат все одно не буде задовольняти умови, які диктує ДСТУ по оформленню бакалаврської дипломної роботи.

Після того, як було розглянуто команди для поділу тексту по частинам, слід розібратись саме з їх налаштуванням. ДЕСТУ 2019 – 2020 року переглянув питання по оформленню оголошень, саме тому були прийняті нові стандарти, які не яким чином не збігаються зі стандартами системи LaTeX. Саме для новітніх стандартів, у системі існує можливість перевизначити існуючи установки в потрібні саме користувачу. Робиться це за допомогою спеціального коду:

```
\usepackage{titlesec}
\titleformat{\chapter}[display]
{\filcenter}
{\MakeUppercase{\chaptertitlename} \thechapter}
{8pt}
{\bfseries}{}
\titleformat{\section}
{\normalsize\bfseries}
{\thesection}
{1em}{}
\titleformat{\subsection}
{\normalsize\bfseries}
{\thesubsection}
{1em}{}
```

Цей код призначений не для всіх типів заголовків, він налаштовує виключно тільки главу, секцію, та підсекцію. Фрагмент коду слід розглядати зверху в низ [40] [1]. Першою йде команда, яка підключає пакет, без якого власне налаштування заголовків зробити не можливо. Далі починаючи з установок для глави можна бачити, що відбувається деяка зміна у синтаксисі запису заголовка, а саме команди роблять його текст жирним, та переводять всі його букви у верхній регістр. А також до стандартного розміру шрифту у системі, додається додатково 8pt. Після заголовка глави йдуть однакові налаштування для секції та підсекції. Вони роблять товщину шрифтів жирними, а їх стиль нормальним.

Також розглянемо доповнення фрагменту кода для оформлення заголовків:

```
\titlespacing*{\chapter}{0pt}{-30pt}{8pt}
\titlespacing*{\section}{\parindent}{*4}{*4}
\titlespacing*{\subsection}{\parindent}{*4}{*4}
```

Перша команда задає нове положення для заголовків глави в pt. Наступні дві команди помножають на чотири, всі параметри абзацного відступу. Виходячи з цієї формули, ми отримуємо шуканий результат. Команда \renewcommand*{\thesection}{\arabic{section}}, скидає значення лічильника, та робить секцію перш пріоритетним заголовком, що дуже спрощує оформлення нумерації секцій та підсекцій у тексті.

Останнім та дуже важливим елементом, який потрібно стилізувати згідно з стандартами є оголошення. Воно потрібно бути оформлено згідно з вимогами ДСТУ, саме тому для нього було розроблено такий фрагмент коду, який вирішує питання правильного оформлення, змінюючи стандарти оформлення у системі. Код для налаштування оголошення:

```
\usepackage{textcase}
```

[1] [40] StackExchange. How to remove “CHAPTER” for unnumbered chapters in ToC? (заг. з екрана). URL: https://tex.stackexchange.com/questions/372177/how-to-remove-chapter-for-unnumbered-chapters-in-toc. (дата звернення 27.12.2019).

```
\makeatletter
\let\oldcontentsline\contentsline
\def\contentsline#1#2{
\expandafter\ifx\csname l@#1\endcsname\l@section
\expandafter\@firstoftwo
\else
\expandafter\@secondoftwo
\fi
{\oldcontentsline{#1}{\MakeTextUppercase{#2}}}
{\oldcontentsline{#1}{#2}}
}
\makeatother
```

Цей код можна назвати навіть скриптом, тому що він містить елементи логічного програмування, його суть полягає у тому, щоб змінити оформлення секцій у оголошенні, тобто зробити, щоб заголовки секції були набрані автоматично великими літерами. Тепер, коли ми вже розглянули багато команд, та блоків коду, які налаштовують документ та його текст, за допомогою преамбули. Ми з вами можемо перейти до команд, які слід використовувати для оформлення самого тексту, нижче буде розглянуто, за допомогою яких саме команд пишеться сам текст у проекті, які команди можуть впливати на нього. Також буде розглянуто як саме у LaTeX, можна створити власноруч команди, а також команди за допомогою яких користувач може задавати списки елементів у тексті.

Почнемо ми з команд \begin{document} та \end{document}. Треба зазначити, що без участі цих двох команд, створення документу просто неможливе. Як ми бачимо, ці команди задаються для розмітки, де починається текст і саме де він закінчується, у фігурних дужках ми бачимо ключове слово document, це значить, що з команди \begin{document} починається сам текст документу, але це не означає що не можна, задавати такі ж, вкладенні конструкції відносно цієї. Тобто ми можемо використовувати саме ці команди в тексті документу, наприклад для того, щоб ділити сам текст за значенням. Звичайно ж система має можливість установки оголошення, але поділення тексту вище описаним методом не буде зайвим, тому що в системі можна зв’язувати елементи один з одним. Як було сказано раніше, LaTeX передбачає можливість зміни тексту за

допомогою певних команд. Існують такі команди, як \sffamily – змінює звичайний шрифт на гарнітуру без засічок, яка також має назву Sans Serif, \ttfamily – широка гарнітура (Typewriter), \rmfamily – звичайна гарнітура (Roman). Далі можна привести такі, як \bfseries – включення жирного шрифту, \mdseries- відключення жирного шрифту, та на завершення, а також \itshape – який встановлює курсив, \slshape – похилий шрифт, та \upshape – звичайне начертання.

Далі розглянемо фрагмент коду, завдяки якому була створена в LaTeX титульна сторінка для моєї БКР:

```
\begin{titlepage}
\centerline{МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ}
\centerline{{ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ
УНІВЕРСИТЕТ}}
\vskip1.5cm
\rightline{\uline{Факультет комп’ютерних
наук,}\hspace{23mm}}
\rightline{\uline{управління та
адміністрування}\hspace{23mm}}
\rightline{Кафедра \uline{автоматизованих
систем}\hspace{17mm}}
\rightline{\uline{моніторингу навколишнього
середовища}}
\vskip1.5cm
\centerline{\bf{Бакалаврська кваліфікаційна робота}}
\vskip0.3cm
\centerline{на тему: "< \uline{Електронна розмітка та
програмне кодування структурних}}
\centerline{\uline{елементів дипломних робіт за
допомогою методології TeX}">}
\end{titlepage}
```

Як можна бачити з коду, спочатку було виділено оточення для титульного аркушу, потім першою командою було проведено маніпуляцію з текстом, завдяки якій, його було розміщено строго посеред сторінки. З другою командою відбувається теж саме, так була зроблена шапка титульної сторінки. Далі йде команда \vskip, яка дає змогу зробити відступ від вже оформлених речень, до ще неоформлених. Після цього за допомогою команди \rightline, текст притуляється до правої границі сторінки, там записується інформація про кафедру і так далі. Потім, знову використовується команда пропуску, та \centerline, за допомогою якого, вже

оформлюється тема БКР та основні моменти титульного аркушу. Далі все робить по такий ж самій схемі. Зверху приведено не весь код титульного листу, а лише його фрагмент. Повний код буде представлений у додатку А.

Обов’язково треба розглянути, як саме у системі TeX/LaTeX відбувається підключення рисунків. Робиться це дуже просто, за допомогою таких команд:

```
\begin{figure}[h!]
\centering
\includegraphics[width=0.9\textwidth]{d1}
\caption{Переваги макропакету LaTeX }
\label{d1}
\end{figure}
```

Перша команда розпочинає оточення, для того щоб система розуміла, що річ йде саме про картинку, у фігурних дужках вказується, елемент figure. Далі йде команда, яка відповідає за централізацію елементу посеред сторінки. Наступна установка, задає ширину об’єкту, та указує який саме файл потрібно завантажити в проект. Для цього користувачу потрібно всього то додати потрібну картинку, у папку проекту, та вказати вірний путь до файлу у фігурних дужках. Команда \caption потрібна для завдання підпису до картинки. Після всіх цих команд слідує остання, яка задає ідентифікацію об’єкту, для того, щоб була можливість провести зв’язок між елементами системи.

Для того, щоб додати виноски у текст, треба скористуватись такою командою, як \footnote. Тест виноски потрібно вставити у фігурні дужки команди, і він одразу виведеться внизу сторінки під лінією, яка розділяє основний текст, та текст виноски. Це дуже легке рішення, важливого питання, яке не вирішується так просто в інших редакторах не зв’язаних з LaTeX.

Далі будемо розглядати такий важний розділ в системі, як формули, та їх оформлення. У системі LaTeX, для оформлення формул існують деякі правила. Як можна було вже здогадатись вони задаються за допомогою

певних команд, які підключаються до системи разом з деякими пакетами. Система відрізняється від інших, як раз таки якістю оформлення формул, тому що система комп'ютерної верстки TeX, була придумана перш за все для роботи з формулами, чому і отримала таку шалену популярність у математиків та вчених [41] [1)].

Розглянемо код для оформлення формул детальніше. Для цього використовують знак $, наприклад, набираємо $ Y '= f (x) $, отримуємо y'= f (x). Якщо рядок закінчується знаком = , + , – , або подібним, то знак потрібно продублювати і на наступному рядку. Для цього використовують команду \hm, наприклад, з $ Y (x) = x ^ 2 \hm -2x \hm + 3 $ отримуємо y (x) = x2 - 2x + 3, або y (x) = x2 - 2x + 3. Слід позначити що вони можуть бути 2-х видів, тобто можуть знаходитись як у самому тексті, так і виноситись в окремий ряд по за текстом [42] [2)]. Якщо цікава нам формула повинна розташовуватись безпосередньо у тексті, то за правилами LaTeX'у , її написання повинне починатись з символу $. Існують випадки коли користувачу потрібно винести саму формулу поза текст, для цього в синтаксисі документу, йому потрібно окружити формулу знаками $, по одному символу з заду та переду. Дуже цікавий факт, що в даній редакторській системі, пробіли які задає user, не зчитаються LaTeX'ом [43] [3)]. Вони ставляться у формулі безпосередньо так, як вирішить система. У неї є певні стандарти щодо формул, які були встановлені розробниками з метою покращення їх якості та наявності у тексті. Оскільки процес перетворення коду у текст повністю автоматизований, користувачу потрібно просто натиснути на кнопку компіляції, та очікувати доки LaTeX, зверстає сторінку з текстом та формулами. Слід позначити що кожен символ у формулі ,яка знаходиться у

[1)] [41] Глонти Н.Л., Грицаенко И.А. TEX в ИФВЭ 5. Шрифты и работа с ними. Препринт ИФВЭ 92–127, Протвино, 1992. 145 с.

[2)] [42] Computation with Ablosulety No Spaxe Overhead (заголовок з екрану). URL: http://kappa.cs.karelia.ru/~olbgvl/ktmath/beamerexample1.pdf (дата звернення: 24.01.2020).

[3)] [43] Богоявленская О. Ю. Компютернные технологии в математике (загол. з екрану). Петрозаводськ, 2015. URL:http://kappa.cs.karelia.ru/~olbgvl/ktmath/matem-skt.html (дата звернення: 24.09.2019).

контейнері $..$, представляє собою ім’я зміної , та пишеться на виході не звичайним курсивом, а математичним. Цей шрифт йде у пакетах які підключаються у преамбулі додатково, якщо потрібно працювати з формулами.

Далі розглянемо пакети які потрібно підключити для роботи з формулами: \usepackage(mathtext) – цей пакет дає користувачу змогу, написати пояснення для будь якої формули у документі, на підключеній їм мові, будь то українська, російська, чи інша.

Далі йде обов’язковий пакет, без якого використання формул неможливе: \usepackage [amsmath, amsfonts, amssymb, amsthm, mathtools] – у цьому пакеті є 5 атрибутів, пройдемося по кожному з них (amsmath- дає можливість зручної верстки , складних та дуже складних формул, а також регулює, та масштабує текст формул, та прорисовує формули у рамках, amsfonts – відповідає за загрузку альтернативних шрифтів, наприклад готичного чи ажурного, на випадок коли для об означення елементів формул не буде вистачати латинського алфавіту, amssymb – доповнює підключені шрифти сотнями додаткових математичних символів, amsthm – додає окруження , наприклад “ теорема ”, mathtools – підключає багато різних математичних інструментів з метою поліпшення можливості маніпуляції з формулами.)

\usepackage [icomma] – підключає розумну кому, тобто дає системі розуміння про те , стоїть звичайна кома, чи кома від десятинної дроби.

\usepackage[euscript] – підключає шрифт Евкліда.

Далі будемо розглядати команди , які безпосередньо поліпшують якість роботи с формулами, команди за які LaTeX і має свою популярність: \newcommand*{\hm}{1}{#1\nobreak\discretionary{}{\hbox[$\mathsurround=0pt #15]}} – відповідає за перенос символів та знаків на нову строчку у формулі.

\setlength{\mathsurround}{2pt} – робить так щоб кожна формула у тексті була оточена відступами у 2pt, це значення можна змінювати по бажанню, система буде адаптувати текст під ваші налаштування.

За допомогою деяких команд, можливо самостійно змінювати номера елементів документу, таких, як: сторінка, формула чи малюнок. Але є пакет який дає ті ж самі можливості, але тільки для формул, цей пакет називається amsmat. Якщо в преамбулу документу вписати: \number within {equation} {section} – то в кожній секції рівняння нумеруватимуться незалежно один від одного.

Усередині оточення рівняння, які повинні розміщуватися на окремих рядках, відокремлюються одна від одної командами \\ . Вираз в межах одного рядка має складатися з трьох частин (можливо порожніх), розділених амперсантом &. Кожна частина поміщається в свій стовпець. У лівому стовпчику формули притискаються до правому краю, в середньому – центруються, а в правій колонці, притискаються до лівого краю. Іншими словами, символи & задають точки вирівнювання рівнянь з різних рядків по вертикалі. Кожен рядок в оточенні eqnarray отримує при друку свій номер. Припинити нумерацію будь-якого рядка можна командою \nonumber [44] [1]. Таким чином відбувається створення посилань у LaTeX. Допустимо a и b – цілі числа. Виявим що c:

```
\begin{equation}\label
{prostaya_formula}
c=a+b
\end{equation}
```

Тоді c, заданне формулой (\ref{prostaya_formula}), також являється цілим числом. Слід відмітити, що в системі LaTeX, хоч і є можливість зробити проект, який буде повністю розташовуватись у одному файлі, однак це не розсудливо, тому що, користувачу буде дуже складно виправляти помилки, навіть якщо він розділяв блоки коду коментарями.

---

[1] [44] Глонти Н.Л., Грицаенко И.А.. Многоязычный LaTEX. Препринт ИФВЭ 94–57, Протвино, 1994. 161 с.

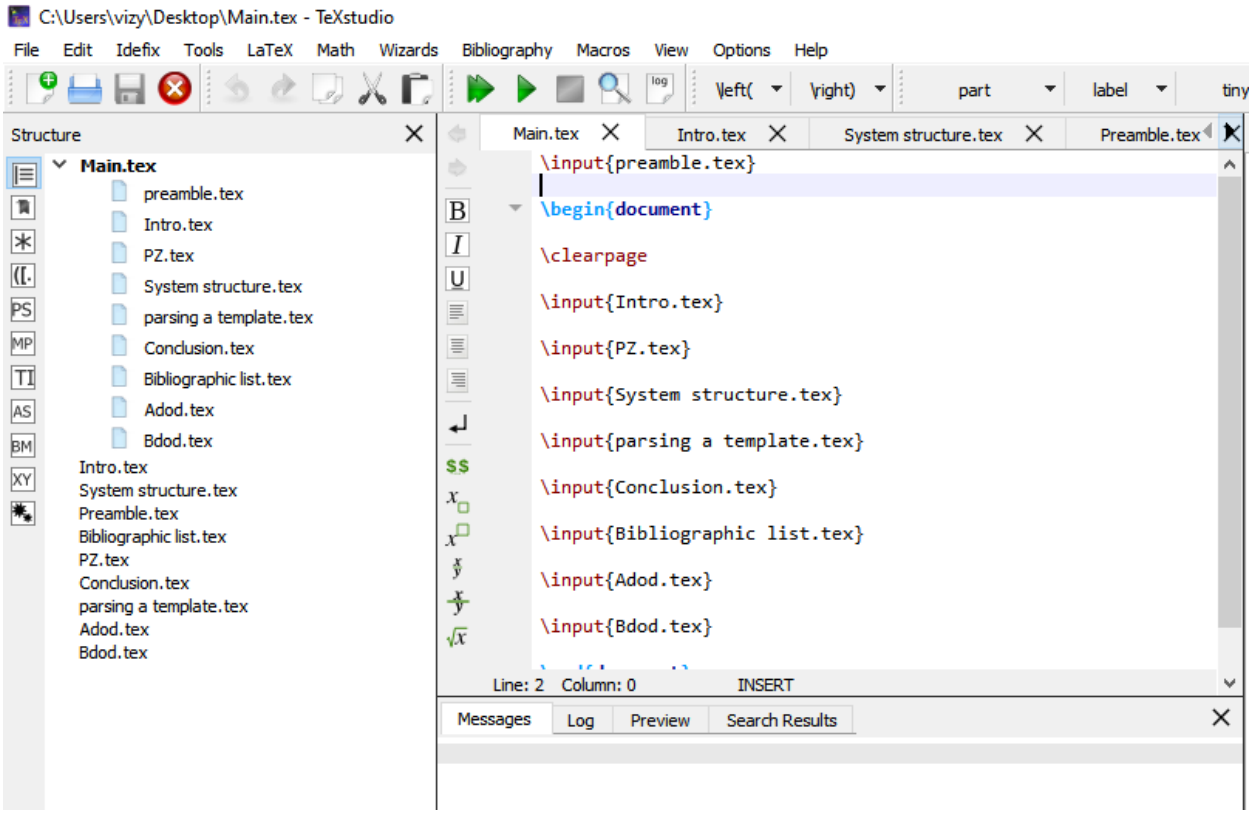


Рисунок 23 – Структурне підключення додаткових файлів.

Слід робити так, ми створюємо основний файл, і в цей файл підключаємо всі додаткові файли, кожний із яких має окремий код, який керує деякою частиною дипломного проекту. Після підключення всі файли зв’язуються один с одним по порядку, тобто перший підключений файл, буде мати більш пріоритетне місце розміщення, ніж файл , який підключений другим чи третім, і так по черзі, доки всі фали не будуть підключені до основного. Зверху наведено приклад у виді скриншоту, який був взятий з мого дипломного проекту (рис.23).

На цьому рисунку ми можемо побачити приклад, як само треба структурувати файли у системі LaTeX. Як ви можете бачити, кожен файл виконує окрему роль, ці ролі буде детально описано нижче. Користувач може підключати стільки файлів, скільки йому треба для того, що реалізувати свій проект. В мене у дипломному проекті до основного файлу main.tex, підключаються такі файли, як бібліографічний список, преамбула, введення

та багато інших. Знизу приведений блок коду, який відповідає за підключення всіх цих файлів, а також приведений опис кожного з них.

```
\documentclass[a4paper,14pt]{extarticle}
\input{preamble.tex}
\begin{document}
\clearpage
\input{Intro.tex}
\input{PZ.tex}
\input{System structure.tex}
\input{parsing a template.tex }
\input{Conclusion.tex}
\input{Bibliographic list.tex}
\input{Adod.tex}
\input{Bdod.tex}
\end{document}
```

Перш за все, до головного файлу, підключається преамбула. Її суть полягає в тому, щоб завдати налаштування усім іншим файлам, які будуть підключатись після неї, а також і для налаштування самого проекту. Після преамбули, ми зазвичай підключаємо саме той файл, з якого почалась робота над моїм дипломним проектом, звичайно мова йде про введення, яке з'єднується з основним файлом через команду – \input{Intro.tex}. Далі йде файл, який містить дипломне завдання, в розгорнутому виді, та основна література, яка допомагає читачу ознайомитись з LaTeX, дізнатись про історію його створення, та особливості [45] [1)]. Це дуже важливо для того, щоб користувач мав чітку уяву про те, що із себе представляє дана система. Після цього в моєму дипломному проекті йде аналіз, саме актуальності теми мого диплому. Там досконально описано, чому у наш час треба користуватись LaTeX, розписані всі недоліки та плюси, яких як показує практика набагато більше. У файлі також є порівняння цієї системи з іншими. Далі йде файл який надає ще більше інформації о проекті та розкриває його з різноманітних кутів. Ще є файли які містять інструкцію для користувачів, щодо роботи системи, а саме про те, у якому форматі зберігаються файли, як

[1)] [40] Tobias Oetiker. Не очень краткое введение в LATEX2e (Или LATEX2e за 94 за минуты). 1999 год. 108 стр.

встановити редактори LaTeX, як зв’язати файли у системі і т.д. Одним з останніх файлів є той, який містить у собі заключення. Заключення було проведено аналізуючи всю виконану роботу, а також її актуальність. Після нього підключається бібліографічний список, який містить посилання на текст документу, а також описує джерела з яких було взято інформацію. Закінчується мій структурний план підключення файлів, додаванням у головний файл, додатків А та В.

ВИСНОВКИ

Виходячи з інформації, яка надана у контексті даної бакалаврської кваліфікаційної роботи, можна зробити висновок, що система комп'ютерної верстки TeX, та його макро пакет LaTeX (який надає системі унікальні властивості) являються найкращим варіантом, серед редакційних систем і процесорів на сьогоднішній день. Це обумовлено багатьма факторами. Найважливішими з них являються якості, які дозволяють системі виконувати TeX – код на будь – якій, існуючий системі, це означає що код є повністю неприв'язаним не до процесорів, та ОС. Цей факт дає виробникам змогу з упевненістю називати свій продукт, як мінімум гнучким. Ще одним плюсом системи TeX\LaTeX, є те що вона повністю автоматизована. Це означає, що будь – який користувач, може за бажанням використовувати вже розроблений їм макет, скільки завгодно разів. Система має свої лічильники, які треба підключати до документу для того щоб, коли буде добавлятись новий елемент в умовний проект, він мав свій порядковий номер, який автоматично присвоїть йому LaTeX. На цьому розбір автоматизованості проекту в системі не закінчується, але у самій БКР вже були розібрані всі її приклади та умови.

Також слід зазначити, що на сьогоднішній день існує безліч різноманітних LaTeX – редакторів, завдяки яким користувачу можуть повністю зануритись у контент – частину свого документа, при цьому зовсім незважаючи уваги на оформлення та дизайн тексту. Річь у тому що система сама, виходячи з призначених для користувача налаштувань, які записуються їм у преамбулі документу, компілює та оформлює дизайн документу.

Слід також сказати про те, що все програмне забезпечення, яке відноситься до LaTeX'у, є повністю безкоштовним, а саме тому вони також є й ПЗ з відкритим вихідним кодом. Ця властивість одразу вирішує два питання. Прибирає проблему піратства, а також дає смогу користувачам

самостійно оновлювати систему, та писати для неї свій код, який буде доповнювати систему тим, чи іншим чином. Це дуже вигідна стратегія саме для виробників LaTeX.

Система LaTeX є дуже популярною в західних країнах, але в країнах СНГ вона не користується таким шаленим попитом. Можна зробити висновок, що цей фактор обумовлений тим, що по – перше мало людей знає про її існування, а по – друге тим що всі користувачі, які кожен день стикаються з оформленням документів, настільки сильно звикли до MS Word, що просто не готові замінити його, навіть більш кращою редакційною системою.

LaTeX дозволяє працювати з формулами найкращим образом, не використовуючи для цього додаткові програмні забезпечення. Саме тому науковці всього світу використовують його для оформлення своїх робіт. Якщо взяти наприклад інші редакційні системи, то можна зробити висновок, щодо якості надрукованих формул з їх допомоги.

Система TeX/LaTeX це вибір на користь прогресу і на мою думку, його слід у скорому часі популяризувати саме в Україні. Цей крок дасть змогу сучасним користувачам оформлювати свої документи по стандартам ДЕСТ, при цьому витрачаючи мінімум часу на те, щоб виправляти помилки інших систем. Також система дозволить своїм користувачам, використовувати ще безліч можливостей, а також розробляти для неї нові можливості, шляхом розробки нових, та різноманітних шрифтів.

Система досить складна для вивчення, і в неї є деякі недоліки, однак вона дає набагато більше корисних можливостей, завдяки яким можна закрити очі на всі мінуси системи, та бачити в ній тільки плюси. Незабаром версія LaTeX буде оновлена, і виробники порадують нас ще більшою автоматизованістю і гнучкістю, але і без цього оновлення система TeX/LaTeX є найкращім рішенням по роботі та оформленню документів і проектів, а також з їх друкуванням.

ПЕРЕЛІК ПОСИЛАНЬ

1. Kopka H., W.Daly. P. A guide to LaTeX: document preparation for beginners and advanced users (навч. посібник). 3rd ed.p.cm. London : Addison–Wesley, 1999. 609p.
2. Goossens M, Mittelbach F, Braams J., Carlisle D., Rowley C. The LaTeX Companion (навч. посібник). 2nd ed.p.cm. Boston: Addison–Wesley, 2004. 1119p.
3. Kottwitz S. Latex Beginer's Guide (навч. посібник). Olton: Packt Publishing, 2011. 336p.
4. Роженко А. Искусство верстки в LaTeX'е (навч. посібник). Новосибирск: Изд.ИВМиМГ СО РАН, 2005. 398с.
5. Балдин Е. Компьютерная типография LaTeX (навч. посібник).. Санкт-Петербург: БХВ Петербург, 2008. 304с.
6. LaTeX – Wikipedia (заголовок з екрану). URL: https://ru.wikipedia.org /wiki/LaTeX. (дата звернення: 19.04.2020).
7. Microsoft Word та Latex: снова вместе (заголовок з екрану). URL: https://https://itc.ua/articles/microsoft_word_i_latex_teper_vmeste_27030 (дата звернення: 20.04.2020).
8. ShareLaTeX, Онлайн редактор LaTeX (заголовок з екрану). URL: https://ru.sharelatex.com (дата звернення: 19.04.2020).
9. Обзор редакторов для работы в LaTeX (заголовок з екрану). URl: http://mydebianblog.blogspot.com/2013/02/latex-editors-aintegratlatex.html (дата звернення: 23.04.2020).
10. Установка и настройка программ для редактирования LaTeX файлов (заголовок з екрану). URL: http://blog.harrix.org/article/849. (дата звернення: 23.04.2020).
11. Установка PSCyr для Latex (заголовок з екрану). URL : http://harrix. org/ article/444 (дата звернення: 23.04.2020).

12. Степанова Е.Н. Издательская система LaTeX (навч.посібник). Петрозаводск: Издательство Петрозаводского университета ,1999. 155с.
13. Львовский С. М. Набор и верстка в пакете LATEX (навч.посібник). Москва: Космоинформ, 1995. 374 с.
14. Гуссенс М. Путеводитель по пакету LaTeX и его графическим розширениям (навч. посібник). Москва: Мир, 1999. 606 с.
15. Богоявленская О.Ю. LaTeX для математиков (навч.посібник). Петрозаводск: ПетрГУ, 2016. 26с.
16. Douglas Downing and Michael A. Covington. Dictionary of Computer and Internet Terms. 8th ed. Barron's Educational Series. New York: Woodbury, 2003.592p.
17. The beamer class. User Guide for versio 3.23. (загол. з екрана). URL: http://bitbucket.org/rivanvx/beamer (дата звернення : 20.11.2019).
18. Guidelines for the use of Beamer (загол. з екрана). URL: http://www1. uwindsor.ca/math/sites/uwindsor.ca.math/files/08-02.pdf. (дата звернення 20.11.2019).
19. Lamport L. Latex: A Document Preparation System (навч. посібник).2 nd. ed. London: Addyson–Wesley,1994. 272p.
20. Knyt D.TeX for beginner (навч. посібник). 1 st. ed. New-Yourk: Williams, 1993. 560с.
21. И. А. Котельников, П. З. Чеботаев. Издательская система LATEX 2ε (навч. посібник). Новосибирск: Сибирский хронограф, 1998.496с.
22. П. Каров. Шрифтовые технологии. Описание и инструментарий. Москва: Мир, 2001. 454 с.
23. D. E. Knuth. The TEXbook, часть A серии Computers and Typesetting (навч. посібник) . New-Yourk :Addison-Wesley, 1984. 496p.
24. М. Спивак. Восхитительный TEX: руководство по комфортному изготовлению научных публикаций в пакете AMS-TEX (навч. посібник). Москва: Мир, 1993.282с.

25. Столяров А.В. Сверстай диплом красиво: LATEX за три дня (навч. посібник). Москва: МАКС Пресс, 2010. 100 с.
26. Говорухин В., Цибулин Б., Компьютер в математическом исследовании: Maple, MATLAB, LaTeX (навч. посібник). Санкт-Петербург: Питер, 2001. 624 с.
27. G. Gratzer. First steps in LATEX (навч.посібник). Birkhauser: Springer-Verlag, 1999.131p.
28. Богоявленская О. Ю. Компютернные технологии в математике (загол. з екрану). Петрозаводськ, 2015. URL: http://kappa.cs.karelia.ru/~ olbvgl ktmath/matem-skt.html. (дата звернення: 24. 09. 2019).
29. GitHub. Amet13/bachelor-diploma (загол. з екрана). URL https:// github.com/amet13/bachelor-diploma/inc/peamble.tex. (дата звернення: 27.12.2019).
30. Диплом бакалавра в LaTeX в 150 строк (загол. з екрана). URL: https://www.pvsm.ru/diplom/8387/print/. (дата звернення 23.11.2019).
31. Диплом бакалавра в LaTeX, или ДСТУ 3008-95 в 150 строк (загол. З екрана). URL: https://habr.com/ru/post/144648/. (дата звернення: 24.11.2019).
32. GitHub. Aoklyunin / CSI_REPORT (загол. з екрана). URL: https://github .com/aoklyunin/CSI_REPORT/blob/master/arhive/Заключение.tex. (дата звернення: 24.12.2019).
33. Linux.Org.Ru. Настройка библиографии LaTeX (заг. з екрана). URL: https://www.linux.org.ru/forum/general/11477368. (дата звернення: 27.12.2019).
34. GitHub. Vrovachev/diplom (загол. з екрана). URL: https://github.com/ vrovachev/diplom/blob/master/gost.tex. (дата звернення: 25.12.2019).
35. StackExchange. How to change ampersand's font in formula? (заг. з екрана). URL:https://tex.stackexchange.com/questions/310517/how-to-chan geampers-ands-font-in-formula. (дата звернення 27.12.2019).

36. Глонти Н.Л., Грицаенко И.А. TEX в ИФВЭ 5. Шрифты и работа с ними. Препринт ИФВЭ 92–127, Протвино, 1992. 145 с.
37. Computation with Ablosulety No Spaxe Overhead (заголовок з екрану). URL: http://kappa.cs.karelia.ru/~olbgvl/ktmath/beamerexample1.pdf. (дата звернення : 24.09.2019).
38. G.Gratzer. Math into TEX. A simple introduction to AMS-LATEX (навч. посібник) .Birkhauser, 1993.294с.
39. Глонти Н.Л., Грицаенко И.А.. Многоязычный LaTEX. Препринт ИФ Вє 94–57, Протвино, 1994. 161 с.
40. Tobias Oetiker. Не очень краткое введение в LATEX2e (Или LaTEX2e за 94 минуты). 1999 год. 108 с.
41. Великодний С. С., Нямцу К. Є. Сучасна інформаційна система дляпідготовки LaTeX-документів «TeXstudio». Інформатика, інформаційнісистеми та технології: тези доп. 16-ї Всеукр. конф. студ. і мол. наук. Одеса, 19 квітня 2019 р. Одеса, 2019. С. 11–13.
42. Великодний С. С., Тимофєєва О. С., Зайцева-Великодна С. С., НямцуК. Є. Порівняльний аналіз властивостей відкритого, вільного такомерційного програмного забезпечення. Інформаційні технології такомп’ютерна інженерія. 2018. № 1 (41). С. 21–27.
43. Великодний С. С., Тимофєєва О. С., Зайцева-Великодна С. С. Методрозрахунку показників оцінки проекту при виконанні реінжинірингупрограмних систем. Радіоелектроніка, інформатика, управління. 2018. №4. С. 135–142. DOI: 10.15588/1607-3274-2018-4-13.
44. Великодний С. С. Метод представлення оцінки реінжинірингу програмних систем за допомогою проектних коефіцієнтів. Сучасний станнаукових досліджень та технологій в промисловості. 2019. № 1 (7). С. 34–42. (кат. «Б») DOI: https:// doi.org/10.30837/2522-9818.2019.7.034.
45. Великодний С. С. Методи реінжинірингу програмних систем. Технологии приборостроения. 2014. Спец. вып. С. 65–68.

ДОДАТОК А

Фрагменти лістінгів TeX-коду експериментальної програмної реалізації

А.1 Вхідні налаштування преамбули у системі TeX

```
\usepackage[utf8]{inputenc}
\usepackage[ukraineb]{babel}
\usepackage{vmargin}
\setpapersize{A4}
\usepackage{setspace}
\setmarginsrb{3cm}{2cm}{1.5cm}{2cm}{0pt}{0mm}{0pt}{13mm}
\usepackage{indentfirst}
\usepackage{cmap} \usepackage[T2A]{fontenc}
\usepackage{pscyr}
\usepackage{graphicx}
\usepackage{amssymb,amsfonts,amsmath,amsthm}
\usepackage{float}
\usepackage{lipsum}
\usepackage{listings}
\lstset{basicstyle=\sffamily}
\linespread{1.5}
```

А.2 Завдання налаштувань преамбули для інтервалів виносок та шрифтів, а також для оформлення бібліографії

```
%Інтервал виноски та шрифт
\def\setnotebodyfont{\switchtobodyfont[12pt]
\setupinterlinespace[1.0]}
\usepackage[bottom] { footmisc }
\setlength{\skip\footins}{1cm}
\makeatletter
\def\@makefnmark{\hbox{\@textsuperscript{\normalfont\@thefn
mark)}}}
\makeatother
\usepackage{etoolbox}
\makeatletter
\patchcmd{\@makefntext}{1em}{35.5pt}{}{}
\patchcmd{\@makefntext}{1.8em}{35.5pt}{}{}
\makeatother
```

```
%Налаштування преамбули
\makeatletter
\renewenvironment{thebibliography}[1]
{\section*{\refname}
\list{\@biblabel{\@arabic\c@enumiv}}
{\settowidth\labelwidth{\@biblabel{#1}}
\leftmargin\labelsep
\itemindent 16.7mm
\@openbib@code
\usecounter{enumiv}
\let\p@enumiv\@empty
\renewcommand\theenumiv{\@arabic\c@enumiv}
}
\setlength{\itemsep}{0pt}
}
\makeatother
\setcounter{page}{4}
```

А.3 Завдання налаштувань для списків та нумерації сторінок

```
% Списки
\usepackage{enumitem}
\setlist[enumerate,itemize,description]{leftmargin=12.5mm}
\renewcommand{\rmdefault}{ftm} % Times New Roman
\frenchspacing
\usepackage[normalem]{ulem} % для подчёркиваний uline
\ULdepth = 0.16em % расстояние от линии до текста выше/ниже
% Нумерация страниц справа сверху
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\fancyhead[R]{\textrm{\thepage}}
\fancyheadoffset{0mm}
\fancyfootoffset{0mm}
\setlength{\headheight}{17pt}
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0pt}
\fancypagestyle{plain}{
\fancyhf{}
```

```
\rhead{\thepage}
}
\renewcommand{\footnoterule}{\raggedright\rule{4cm}{0.4pt}\
vspace*{0.6ex}}
```

А.4 Завдання налаштувань для списків та нумерації сторінок

```
%Подписи рисунков
\usepackage[tableposition=top]{caption}
\usepackage{subcaption}
\DeclareCaptionLabelFormat{gostfigure}{Рисунок #2}
\DeclareCaptionLabelFormat{gosttable}{Таблица #2}
\DeclareCaptionLabelSeparator{gost}{~---~}
\captionsetup{labelsep=gost}
\captionsetup[figure]{labelformat=gostfigure}
\captionsetup[table]{labelformat=gosttable}
\renewcommand{\thesubfigure}{\asbuk{subfigure}}
\parindent 1.25cm
%Переопределяем секцию на более высокий уровень
\sloppy
\hyphenpenalty=1000
\clubpenalty=10000
\widowpenalty=10000
\renewcommand*{\thesection}{\arabic{section}}
\usepackage{titlesec}
\titleformat{\chapter}[display]
{\filcenter}
{\MakeUppercase{\chaptertitlename} \thechapter}
{8pt}
{\bfseries}{}
\titleformat{\section}
{\normalsize\bfseries}
{\thesection}
{1em}{}
\titleformat{\subsection}
{\normalsize\bfseries}
{\thesubsection}
{1em}{}
% Настройка вертикальных и горизонтальных отступов
```

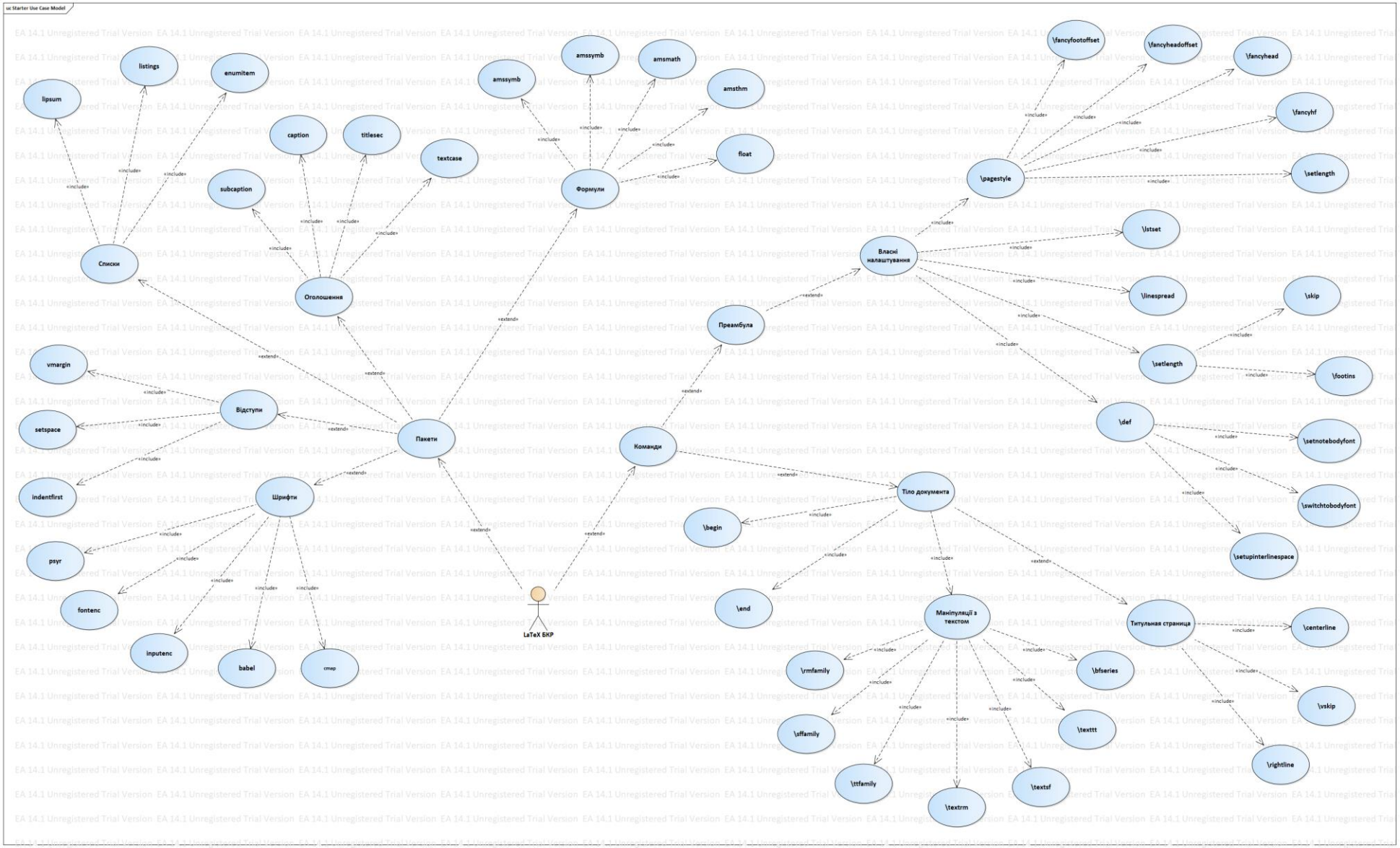
```
\titlespacing*{\chapter}{0pt}{-30pt}{8pt}
\titlespacing*{\section}{\parindent}{*4}{*4}
\titlespacing*{\subsection}{\parindent}{*4}{*4}
```

А.5 Завдання налаштувань для заголовків

```
% Заголовки секций в оглавлении в верхнем регистре
\usepackage{textcase}
\makeatletter
\let\oldcontentsline\contentsline
\def\contentsline#1#2{
\expandafter\ifx\csname l@#1\endcsname\l@section
\expandafter\@firstoftwo
\else
\expandafter\@secondoftwo
\fi
{\oldcontentsline{#1}{\MakeTextUppercase{#2}}}
{\oldcontentsline{#1}{#2}}
}
\makeatother
```

А.6 Завдання налаштувань для оголошення

```
\usepackage{tocloft}
\renewcommand{\cfttoctitlefont}{\hspace{0.38\textwidth}\Mak
eTextUppercase}
\renewcommand{\cftsecfont}{\hspace{0pt}}\renewcommand\cftse
cleader{\cftdotfill{\cftdotsep}}
\renewcommand\cftsecpagefont{\mdseries
\setcounter{tocdepth}{3}
```


uc Starter Use Case Model
listings
enumitem
lipsum
amssymb
amsmath
amsthm
amssymb
caption
titlesec
textcase
subcaption
float
Формули
Списки
Оголошення
Пакети
Преамбула
Команди
Відступи
vmargin
setspace
indentfirst
Шрифти
psyr
fontenc
inputenc
babel
cmap
LaTeX БКР
\fancyfootoffset
\fancyheadoffset
\fancyhead
\fancyhf
\pagestyle
\setlength
\lstset
Власні налаштування
\linespread
\skip
\setlength
\footins
\def
\setnotebodyfont
\switchtobodyfont
\setupinterlinespace
Тіло документа
\begin
\end
Маніпуляції з текстом
Титульная страница
\centerline
\rmfamily
\bfseries
\vskip
\sffamily
\texttt
\rightline
\ttfamily
\textsf
\textrm
«include»
«extend»