

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інженерії програмного забезпечення

ПОЯСНЮВАЛЬНА ЗАПИСКА

до бакалаврської роботи
на ступінь вищої освіти бакалавр

на тему: «**РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ
ПІДТРИМКИ СТАБІЛЬНОГО РІВНЯ ЦУКРУ ІНСУЛІНОЗАЛЕЖНИХ
ЛЮДЕЙ МОВОЮ JAVA**»

Виконав: студент 4 курсу, групи ПД-42
спеціальності
121 Інженерія програмного забезпечення
(шифр і назва спеціальності/спеціалізації)

Петренко Е.Д.
(прізвище та ініціали)

Керівник Трінтіна Н.А.
(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Бакалавр»

Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

Негоденко О.В.

“ _____ ” _____ 2023 року

ЗАВДАННЯ
НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТА

ПЕТРЕНКА ЕДУАРДА ДЕНИСОВИЧА

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка програмного забезпечення для підтримки стабільного рівня цукру інсулінозалежних людей мовою Java»

Керівник роботи: Трінтіна Наталія Альбертівна, к.т.н, доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом вищого навчального закладу від «24» лютого 2023 року №26.

2. Строк подання студентом роботи «1» червня 2023 року

3. Вхідні дані до роботи:

3.1 Науково-технічна література з питань, пов'язаних з програмним забезпеченням щодо розробки на мові Java.

3.2 Офіційна документація Oracle.

3.3 Офіційна документація IntelliJ IDEA.

4. Зміст розрахунково-пояснювальної записки

4.1 Аналіз актуальності розроблюваного програмного забезпечення.

4.2 Аналіз наявних аналогів та вибір інструментів.

4.3 Розробка програмного забезпечення.

- 4.4 Тестування програмного забезпечення.
5. Перелік демонстраційного матеріалу
- 5.1 Титульний слайд.
 - 5.2 Мета, об'єкт, предмет, наукова новизна дослідження.
 - 5.3 Актуальність.
 - 5.4 Аналіз аналогів.
 - 5.5 Технічні завдання.
 - 5.6 Програмні засоби та інструменти реалізації.
 - 5.7 Розробка архітектури.
 - 5.8 Реалізація програми.
 - 5.9 Висновки.
 - 5.10 Апробація результатів дослідження.

6. Дата видачі завдання «25» лютого 2023 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	25.02.2023	Виконано
2	Дослідження аналогів та актуальності додатку	15.03.2023	Виконано
3	Аналіз та вибір інструментів для розробки додатку	23.03.2023	Виконано
4	Проектування та реалізація	05.04.2023	Виконано
5	Вступ, висновки, реферат	30.04.2023	Виконано
6	Розробка обов'язкових демонстраційних матеріалів	08.05.2023	Виконано
7	Попередній захист роботи	19.05.2023	Виконано
8	Здача роботи	01.06.2023	Виконано

Студент

_____ (підпис)

Петренко Е.Д.

(прізвище та ініціали)

Керівник роботи

_____ (підпис)

Трінтіна Н.А.

(прізвище та ініціали)

РЕФЕРАТ

Текстова частина бакалаврської роботи с. 58, 8 табл., 46 рис., 15 джерел.

JAVA, SDK, ORACLE, INTELLIJ IDEA, GUI, GUILDGLUCOSE, ДІАБЕТ, ВУГЛЕВОДИ, ІНСУЛІН, РІВЕНЬ, ЦУКРУ, МЕТОД, КЛАС, БІБЛІОТЕКА, ГРАФІЧНИЙ, ІНТЕРФЕЙС.

Об'єкт дослідження – процес підтримки стабільного рівня цукру інсулінозалежних людей.

Предмет дослідження – програмне забезпечення для підтримки стабільного рівня цукру.

Мета роботи – спрощення процесу визначення дози інсуліну в залежності від спожитої їжі, рівня цукру та індивідуального коефіцієнту.

З урахуванням аналізу процесу контролю рівня цукру та дозування інсуліну у людей з інсулінозалежним діабетом та існуючих засобів для цього можна сформулювати вимоги до майбутнього програмного продукту.

Для платформи Desktop необхідно створити програму, яка зможе виконувати наступні завдання - розрахунок рекомендованої дози інсуліну. Та надавати інші необхідні функції:

- врахування індивідуального коефіцієнту в підрахунках;
- розрахунок дози інсуліну враховуючи рівень цукру, та за потреби додати дози за-для зниження цукру;
- можливість рахувати дозу інсуліну вибираючи їжу зі списку запропонованих страв;
- можливість в разі помилки видалити зі списку страву;
- можливість порахувати дозу інсуліну використовуючи в підрахунках вуглеводи або хлібні одиниці.

ЗМІСТ

ВСТУП.....	10
1. АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ.....	12
1.1 Різниця між типами діабету.....	12
1.2 Симптоми діабету	14
1.3 Ускладнення цукрового діабету	15
1.4 Хронічні ускладнення цукрового діабету.....	16
1.5 Гострі ускладнення цукрового діабету	19
1.6 Інсулін	22
1.7 Способи введення інсуліну	23
1.8 Способи вимірювання рівня цукру в крові.....	25
1.9 Аналіз та огляд аналогів	28
1.9.1 Beetis.....	28
1.9.2 DiabTrend	30
1.9.3 mySugr	32
1.10 Аналіз дослідження та встановлення вимог до ПЗ.....	34
2. ОГЛЯД ЗАСОБІВ СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	35
2.1 Вибір середовища програмування.....	35
2.1.1 Visual Studio Code	35
2.1.2 IntelliJ IDEA	36
2.1.3 NetBeans	38
2.2 Вибір бібліотеки для GUI.....	40
2.2.1 AWT.....	40
2.2.2 Swing.....	41
2.2.3 JavaFX.....	42
2.3 Підсумок та остаточний вибір інструментів	44
3. РЕАЛІЗАЦІЯ ТА ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ СТАБІЛЬНОГО РІВНЯ ЦУКРУ ІНСУЛІНОЗАЛЕЖНИХ ЛЮДЕЙ	45
3.1 Функціональні вимоги до програмного забезпечення	45
3.2 Планування розробки програмного забезпечення.....	45
3.3 Визначення функціональних вимог	48

3.4 Алгоритм дій програмного забезпечення	48
3.5 Графічний інтерфейс програми	50
3.6 Функціонал програми	55
4. ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	63
ВИСНОВКИ.....	66
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	68
ДОДАТОК А.....	70

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

GUI - graphical user interface

IDE - integrated development environment

JDK - java development kit

ПЗ – програмне забезпечення

ХО – хлібні одиниці

ВСТУП

Актуальність теми. Цукровий діабет є однією з найбільш поширених хвороб у світі, що ставить перед системами охорони здоров'я та хворими ряд викликів, оскільки вимагає постійного контролю рівня глюкози в крові та дотримання спеціальної дієти.

За даними Міжнародної діабетичної федерації, близько 537 мільйонів дорослих (20-79 років) живуть з діабетом – це кожен десятий дорослий. Прогнозується, що до 2030 року ця цифра зросте до 643 мільйонів, а до 2045 року - до 783 мільйонів. Україна має 1 мільйон 400 тисяч хворих на діабет, а це кожен 30-й українець, з яких більше 190 тисяч пацієнтів є інсулінозалежними, а понад 7,5 тисяч - діти. Проте насправді поширеність цієї хвороби у країні є в 2-2,5 рази вищою. Цукровий діабет займає третє місце за поширеністю серед захворювань після серцево-судинних захворювань та онкологічних хворіб.

Діабет - це хронічне захворювання, яке характеризується недостатньою виробництвом підшлункової залози інсуліну або неефективним використанням цього гормону організмом. Інсулін регулює рівень цукру (глюкози) в крові, і якщо цей рівень не контролюється, може відбутися гіперглікемія - підвищення рівня глюкози в крові. Це може призвести до ураження багатьох систем організму, зокрема, нервової та кровоносної.

Діабет 1-го типу є захворюванням, при якому організм не виробляє достатньо інсуліну, тому щоденно необхідно вводити інсулін. За даними з 2017 року, було зафіксовано 9 мільйонів випадків діабету 1 типу, більшість з яких були в країнах з високим рівнем доходу. На жаль, навіть за наші дні не відомі причини та методи профілактики цього захворювання.

Підсумовуючи написано вище можна сказати, що важливо вводити правильну дозу інсуліну, щоб запобігти ускладнень. Наразі є багато інформації про їжу, а саме скільки вуглеводів в різних продуктах, це може допомогти з виявленням правильної дози. Тому було прийняте рішення створити програмне забезпечення для підтримки стабільного рівня цукру інсулінозалежних людей.

Була поставлена мета зробити програмне забезпечення з інтуїтивно зрозумілим інтерфейсом, та простим в використанні, для розрахунку необхідної дози інсуліну опираючись на введені дані користувача, такі як: рівень цукру в крові, індивідуальний коефіцієнт, спожита їжа.

Аналізуючи мету, було зроблено список задач:

1. Аналіз обраної галузі.
2. Огляд засобів створення програмного забезпечення.
3. Реалізація та проектування програмного забезпечення для підтримки стабільного рівня цукру інсулінозалежних людей.
4. Тестування програмного забезпечення.

1. АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Різниця між типами діабету

Головною відмінністю між діабетом 1-го та 2-го типу є те, що діабет 1-го типу є генетичним захворюванням, яке зазвичай розвивається на початку життя, тоді як діабет 2-го типу пов'язаний зі способом життя та розвивається з часом. При діабеті 1-го типу імунна система атакує та руйнує клітини підшлункової залози, які виробляють інсулін.

Хоча діабет 1-го та 2-го типу мають спільні риси, вони мають істотні різниці у причинах, впливі на здоров'я та методах лікування. Діабет 1-го типу поширений серед 8% осіб з діабетом, тоді як діабет 2 типу вражає близько 90%. Не слід недооцінювати серйозність обох типів діабету, оскільки вони можуть спричинити серйозні ускладнення при підвищенні рівня глюкози в крові. Тому важливо дотримуватись правильного контролю та лікування, незалежно від типу діабету.

Таблиця 1.1 – Різниця між типами діабету

	Перший тип	Другий тип
Що відбувається?	Організм атакує клітини підшлункової залози, а це означає, що вона не може виробляти інсулін	Організм не може виробляти достатню кількість інсуліну або той інсулін, що виробляється, не працює належним чином
Що спричиняє захворювання?	Наразі ми не знаємо, що спричиняє діабет 1 типу	Ми знаємо, що деякі фактори можуть підвищити ризик розвитку цукрового діабету 2 типу, наприклад, вага та етнічна приналежність

Симптоми	Симптоми 1-го типу з'являються швидше	Симптоми типу 2 легше не помітити, оскільки вони проявляються повільніше
Менеджмент	Тип 1 лікується шляхом прийому інсуліну для контролю рівня цукру в крові	Можна контролювати діабет 2 типу різними способами, ніж діабет 1 типу. До них відносяться ліки, фізичні вправи та дієта. Людям з 2 типом також може бути призначений інсулін
Лікування та профілактика	Наразі не існує ліків від 1-го типу, але дослідження продовжуються	ЦД 2 типу неможливо вилікувати, але існують докази того, що в багатьох випадках його можна попередити і ввести в ремісію

Діабет 1 або 2 типу є захворюванням, при якому рівень глюкози (цукру) в крові збільшується. Це спільна характеристика обох типів діабету, проте вони відрізняються у механізмах, якими це відбувається.

Діабет 1-го типу є аутоімунним захворюванням, при якому організм сам атакує і руйнує клітини, що виробляють гормон інсулін, що призводить до втрати можливості виробляти інсулін.

Інсулін необхідний для всіх нас, оскільки він сприяє перенесенню глюкози з крові до клітин нашого тіла, яку ми використовуємо для отримання енергії. Якщо інсулін відсутній, рівень глюкози в крові підвищується, що може бути шкідливим для здоров'я.

Діабет 2 типу має свої відмінності від інших форм діабету. У разі діабету 2 типу, організм не виробляє достатньо інсуліну або інсулін не працює належним чином через інсулінорезистентність. Це призводить до збільшення рівня глюкози в крові, як і в разі діабету 1 типу.

1.2 Симптоми діабету

Основні симптоми діабету при яких треба йти до лікаря та перевіряти рівень цукру в крові:

Ви відчуваєте сильну спрагу, втрачаєте вагу, ви дуже голодні, маєте нечіткий зір, німіють або поколюють руки чи ноги, почуваетесь дуже втомленими, маєте дуже суху шкіру, маєте рани, які повільно загоюються, маєте більше інфекцій, ніж зазвичай.



Рисунок 1.1 – Ознаки та симптоми діабету

Люди з діабетом 1 типу також можуть відчувати симптоми, такі як нудота, блювання або біль у шлунку. Ці симптоми можуть розвиватися протягом кількох тижнів або місяців і можуть бути досить важкими. Діабет 1 типу зазвичай розпочинається в дитинстві, підлітковому або молодому віці, але може з'явитися в будь-якому віці.

Діабет 2 типу може розвиватися повільно і бути безсимптомним у деяких людей. Оскільки симптоми можуть бути незначними, важливо знати фактори ризику діабету 2 типу та звернутися до лікаря для перевірки, якщо ви маєте підозри на це захворювання.

1.3 Ускладнення цукрового діабету

Високий рівень цукру в крові може призвести до ускладнень діабету, які можуть серйозно пошкодити різні частини вашого тіла, включаючи ноги та очі. Але можна вжити заходів, щоб запобігти або затримати багато з цих небажаних наслідків діабету.

Тривалий високий рівень цукру в крові може призвести до серйозного пошкодження кровоносних судин, що може призвести до порушення кровообігу. Це може призвести до втрати чутливості та іншого відчуття в різних частинах тіла. Якщо людина має пошкодження кровоносних судин та нервів в одній частині тіла, з великою ймовірністю стикнеться з подібними проблемами в інших частинах тіла. Наприклад, якщо пошкоджені ноги, то це може призвести до серйозних проблем із серцем. Однак вчасне лікування та профілактичні заходи можуть допомогти запобігти цим ускладненням діабету.

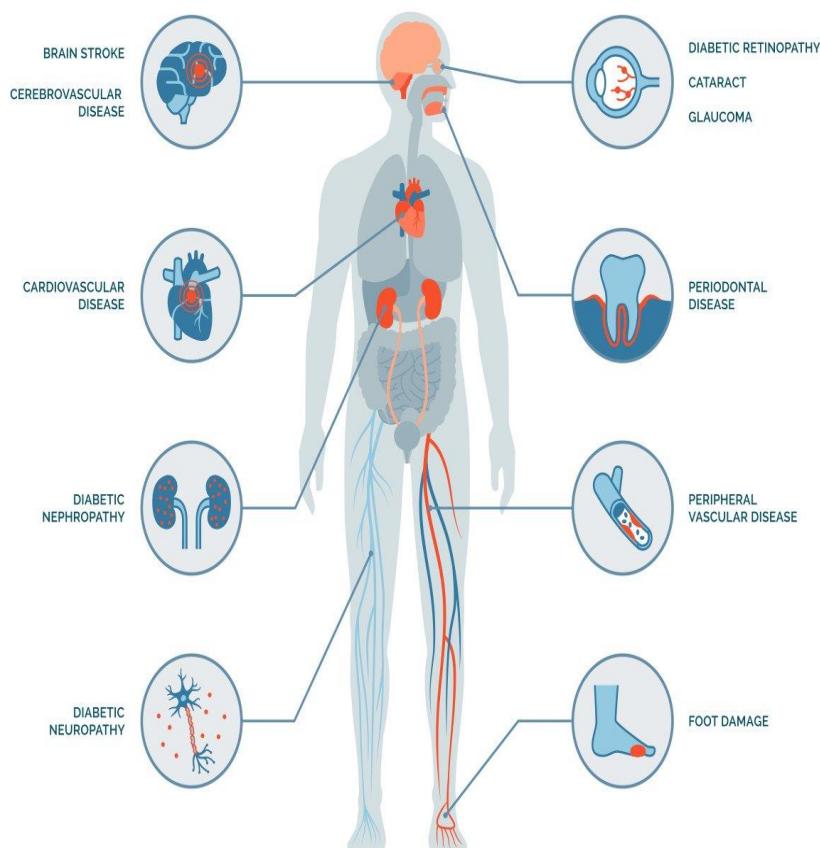


Рисунок 1.2 Ускладнення цукрового діабету

1.4 Хронічні ускладнення цукрового діабету

1.4.1 Проблеми із зором

Діабет може мати різні наслідки для здоров'я очей, а одним з можливих ускладнень є діабетична ретинопатія. Невчасне лікування цієї проблеми може призвести до серйозної втрати зору, що може суттєво вплинути на ваше життя та незалежність. Тому важливо розуміти різні типи діабетичної ретинопатії та знати, як зберегти здоров'я очей. Регулярні візити до офтальмолога допоможуть виявити будь-які проблеми з очима на ранніх стадіях та запобігти втраті зору.

Діабетична ретинопатія - це захворювання очей, до якого більше схильні люди, що живуть з діабетом.

Постійно підвищений рівень глюкози (цукру) в крові та артеріальний тиск можуть серйозно нашкодити кровоносні судини в різних частинах тіла, включаючи очі. Очі містять багато важливих кровоносних судин, які забезпечують доставку крові до сітківки - зони, відповідальної за зорову функцію.

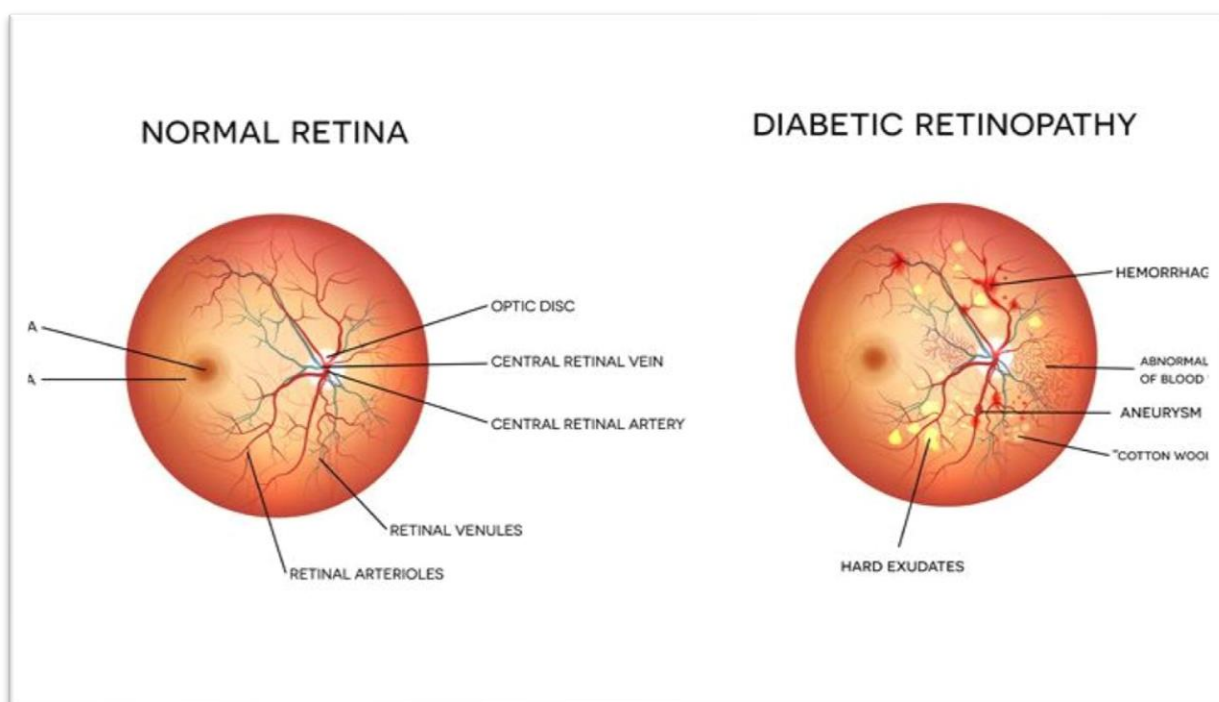


Рисунок 1.3 Діабетична ретинопатія

Високий рівень цукру в крові та артеріальний тиск можуть пошкодити кровоносні судини в очах, що може призвести до блокування, витікання або безладного розростання цих судин. Це може призвести до порушення постачання крові до сітківки очей, що може спричинити проблеми з зором.

Щоб зменшити ризик розвитку серйозних очних проблем, необхідно контролювати рівень цукру в крові та артеріальний тиск. Якщо вже виникли проблеми з очима, то вчасне діагностування та лікування можуть допомогти уникнути довготривалих ускладнень.

Люди з діабетом також мають підвищений ризик розвитку катаракти та глаукоми, що можуть привести до втрати зору. Катаракта зумовлена поступовим затуманенням зору, а глаукома - накопиченням рідини в оці. Як і з діабетичною ретинопатією, підвищений рівень цукру в крові може вплинути на ці стани, тому регулярне проходження офтальмологічних обстежень та контроль рівня цукру в крові є важливими для запобігання розвитку цих проблем з очима.

1.4.2 Проблеми зі стопами

Високий ризик виникнення серйозних проблем зі стопами є поширеною проблемою серед людей з діабетом. Навіть незначне пошкодження стопи може стати проблемою, яка вимагає термінової допомоги. Якщо виявлено високий рівень ризику, необхідно дотримуватися певних заходів для запобігання виникнення серйозних проблем зі стопами та регулярно проводити огляди у лікаря.

1.4.3 Хвороби серця

У людей з діабетом є підвищений ризик серцево-судинних захворювань, які можуть призвести до серцевих нападів та інсультів. Серцево-судинні захворювання також можуть негативно впливати на кровообіг, що може призвести до ускладнень, таких як проблеми з очима та ногами, у людей з діабетом.

Якщо протягом тривалого періоду часу у вас залишається підвищений рівень глюкози в крові, навіть в невеликій мірі, це може спричинити пошкодження кровоносних судин, що може стати причиною серйозних проблем з серцем.

Коли організм не може належним чином використати всю глюкозу, яка перебуває в крові, вона прилипає до еритроцитів і накопичується. Це накопичення може блокувати і пошкоджувати кровоносні судини, що поставляють кров до серця та від нього, зменшуючи доступ серця до кисню та поживних речовин. Як результат, це може призвести до серйозних ускладнень серця.

1.4.4 Діабетична нефропатія (захворювання нирок)

Діабетична нефропатія - це назва ураження нирок, спричинене діабетом. Вона розвивається повільно, протягом багатьох років, і її також називають хворобою нирок.

Приблизно 20% людей з діабетом потребують лікування діабетичної нефропатії, що може викликати стурбованість. Проте, існує багато способів, які можуть допомогти зменшити ризик розвитку цієї ускладнення. Якщо діабетичну нефропатію виявити на ранній стадії, її розвиток можна затримати за допомогою вчасного лікування.

Що спричиняє захворювання нирок? Одне з основних завдань нирок - фільтрувати кров. Вони виводять зайву рідину та продукти життєдіяльності з організму через сечу. Високий рівень глюкози в крові та підвищений кров'яний тиск можуть призвести до пошкодження малих кровоносних судин та фільтрів в нирках.

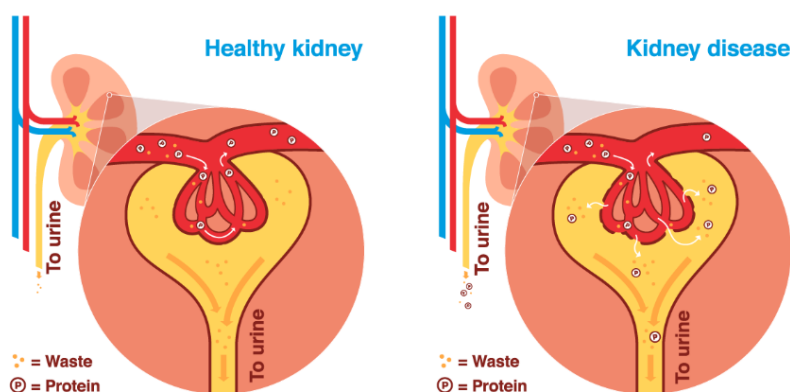


Рисунок 1.4 Діабетична нефропатія

1.5 Гострі ускладнення цукрового діабету

1.5.1 Гіпоглікемія

Гіпоглікемія - це стан, при якому рівень цукру в крові знижується нижче 4 ммоль/л. Виникнення гіпоглікемії може бути пов'язане з неправильним балансом між ліками від діабету, особливо інсуліном, їжею та фізичною активністю. Цей стан не є універсальним для всіх людей з діабетом.

Симптоми у всіх різні, але найпоширенішими симптомами низького рівня цукру в крові є:

- відчуття тремтіння
- відчуття дезорієнтації
- пітливість
- тривожність / дратівливість
- блідість
- прискорене серцебиття
- нечіткість зору
- відчуття голоду
- плаксивість
- втома
- головний біль
- відсутність концентрації уваги

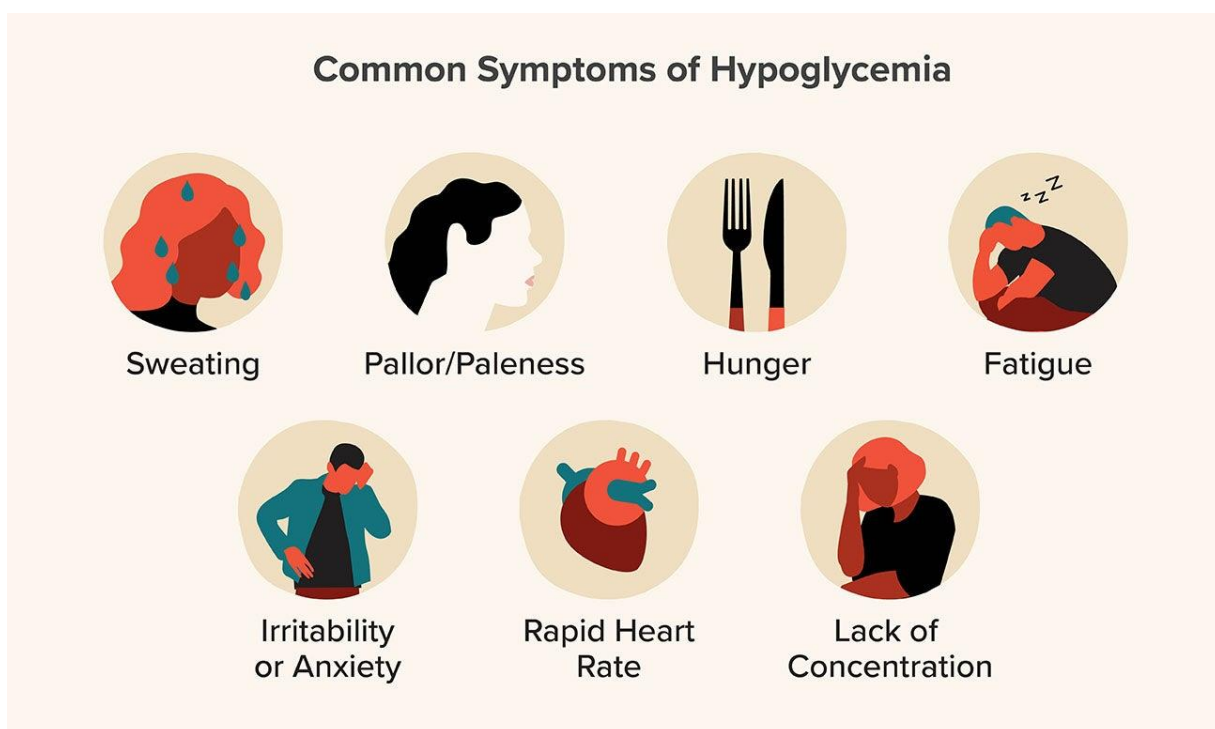


Рисунок 1.5 Симптоми гіпоглікемії

1.5.2 Гіперглікемія

Гіперглікемія - відбувається, коли рівень глюкози (цукру) в крові занадто високий - зазвичай більше 7 ммоль/л натщесерце та більше 8,5 ммоль/л через дві години після прийому їжі. Це стається, коли організм не може виробити достатньо інсуліну для переробки цукру в крові, або не може ефективно використовувати наявний інсулін.

Коли рівень цукру в крові трохи вищий за норму, ви зазвичай не відчуваєте жодних симптомів. Але коли він підвищується, симптоми можуть включати:

- виділення більшої кількості сечі, ніж зазвичай, особливо вночі
- сильну спрагу
- втому і млявість
- молочниця або інші рецидивуючі інфекції сечового міхура та шкіри
- нечіткість зору
- втрата ваги
- нудота

При гіперглікемії симптоми розвиваються повільно і поступово, коли рівень цукру в крові піднімається, тому їх може бути важко помітити. Однак, якщо рівень глюкози в крові залишається високим, симптоми можуть стати дуже виразними і важкими, тому важливо діяти негайно для зниження рівня цукру в крові.

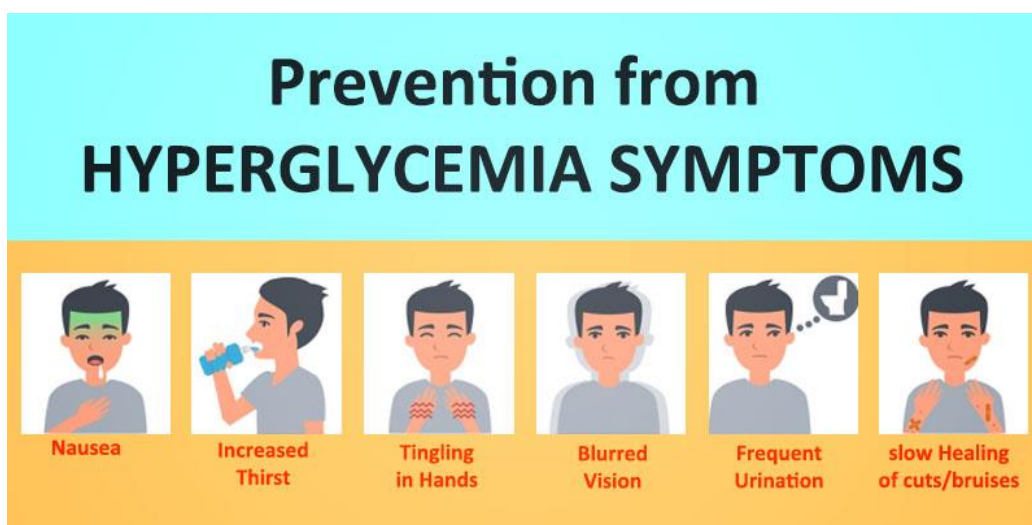


Рисунок 1.6 Симптоми гіперглікемії

1.5.3 Діабетична кома

Діабетична кома - це стан, що може бути фатальним, і він спостерігається виключно у людей з діабетом 2 типу. Його спричинюють значна дегідратація та надмірно високий рівень глюкози в крові.

Гіперосмолярний гіперглікемічний стан (ГГС) є життєво небезпечним станом, який виникає у людей з діабетом 2 типу та характеризується дуже високим рівнем глюкози в крові, зазвичай понад 40 ммоль/л. ГГС може розвиватися протягом декількох тижнів та часто спричинюється поєднанням інших захворювань та зневоднення. Діагноз діабету 2 типу може не бути встановлений у деяких людей до того моменту, коли вони не захворіють на ГГС. Крім того, в окремих випадках у людей з діабетом 2 типу може розвинути діабетичний кетоацидоз (ДКА), але цей стан відноситься до виняткових ситуацій порівняно з ДКА у людей з діабетом 1 типу.

1.5.4 Діабетичний кетоацидоз

Діабетичний кетоацидоз (ДКА) - це гостра ускладнення діабету, яке виникає через нестачу інсуліну в організмі. У такому випадку, організм не може ефективно використовувати цукор для отримання енергії, тому починає розкладати жири. Цей процес супроводжується вивільненням кетонів, які можуть накопичуватися в крові і призводити до її закислення, що називається ацидозом. Якщо не контролювати цей процес, ДКА може стати життєво небезпечним станом.

Діабетичний кетоацидоз є серйозним станом, який відомий тим, що він виникає у людей з діабетом 1-го типу і іноді з діабетом 2-го типу (хоча останні частіше страждають від іншого стану - гіперосмолярного гіперглікемічного стану). У деяких випадках діагноз діабету 1-го типу не ставлять доти, доки не з'являться симптоми ДКА, що може статися з дітьми та дорослими.

У хворих на цукровий діабет найбільш поширеним фактором, що спричинює розвиток діабетичного кетоацидозу, є наявність супутньої інфекції, яка відзначається у 40% випадків. Пропуск введення дози інсуліну є ще одним фактором ризику з відсотковою часткою 25%.

1.6 Інсулін

Ми не можемо жити без інсуліну. Це гормон (хімічна речовина), що виробляється у організмі і допомагає контролювати рівень цукру в крові.

Інсулін також випускається у вигляді ліків, які деякі люди з діабетом приймають, якщо їхній організм не виробляє інсулін - або якщо він виробляється неправильно - так звана інсулінорезистентність.

Усі люди з діабетом 1-го типу приймають інсулін у вигляді ліків. Деякі люди з діабетом 2 типу також приймають інсулін, як і деякі люди з іншими типами діабету, включаючи гестаційний діабет.

Інсулін можна порівняти з ключем, який відкриває двері клітин для глюкози - цукру, який кружляє в крові. Після відкриття дверей клітинами глюкоза може використовуватись для отримання енергії. Інсулін також допомагає організму зберігати надлишок глюкози, щоб не допустити її накопичення в крові.

Інсулін є основним методом лікування для людей з діабетом, який допомагає контролювати рівень цукру в крові та зменшує ризик розвитку ускладнень. Він призначається з метою поліпшення стану хворого та підтримки його здоров'я.

Для контролювання рівня цукру в крові при діабеті одного прийому інсуліну недостатньо. Крім того, необхідно дотримуватися здорового способу життя, включаючи фізичну активність та збалансовану дієту, наскільки дозволяє загальний стан здоров'я. Це сприятиме підтримці рівня цукру в крові в межах необхідного діапазону та зниженню ризику виникнення довгострокових ускладнень.



Рисунок 1.7 Інсулін

Інсулін доступний у формі рідини, яку можна вводити в організм за допомогою ручки та голки, або голки та шприца. Також інсулін можна вводити за допомогою електронного пристрою, який називається інсулінова помпа. Цей пристрій прикріплюється до тіла та забезпечує автоматичне введення інсуліну в організм у відповідності до заданої програми.

Найкраще зберігати інсулін, який не використовується, у холодильнику. Це пов'язано з тим, що інсулін потрібно зберігати при температурі нижче 25°C (77°F). Ідеальна температура зберігання - від 2 до 6°C (36-43°F). Для інсуліну, який використовується протягом дня, зазвичай достатньо кімнатної температури.

1.7 Способи введення інсуліну

1.7.1 Інсулінова помпа

Інсулінова помпа - це невеликий електронний пристрій, який постійно випускає дозу інсуліну, необхідну організму вдень і вночі, що зменшує необхідність у введенні інсуліну шляхом ін'єкцій. Це може допомогти забезпечити належний рівень цукру в крові, але для правильної роботи помпа повинна бути прикріплена до тіла більшу частину часу.

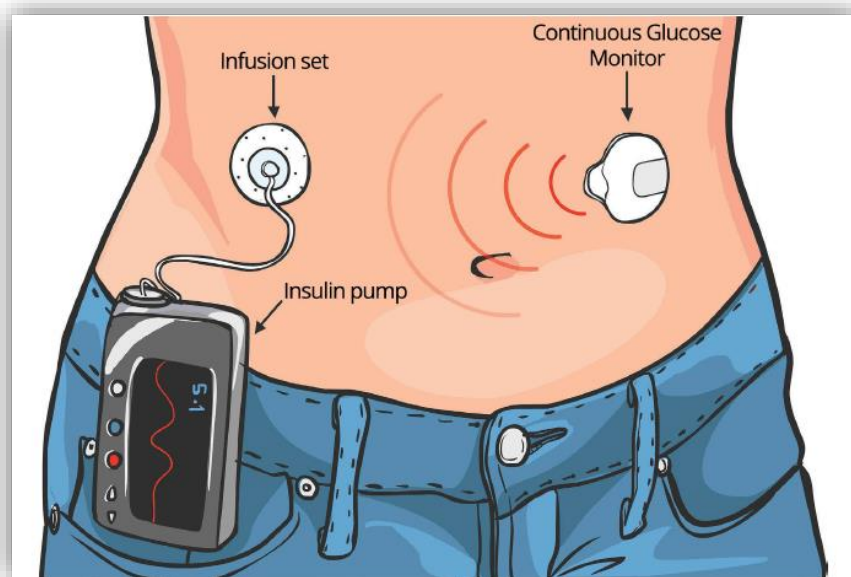


Рисунок 1.8 Інсулінова помпа

Для інсулінової помпи потрібен лише один тип інсуліну, відомий як інсулін швидкої дії.

Для належної роботи інсулінової помпи зазвичай необхідно налаштувати постійний режим подачі невеликих доз інсуліну, що відповідає базальному інсуліну або фоновому рівню. Лікарі можуть допомогти налаштувати дозу відповідно до індивідуального профілю і змінювати її залежно від активності і потреб.

Таблиця 1.2 – Переваги та недоліки інсулінових pomp

Переваги інсулінових pomp	Недоліки інсулінових pomp
Рівень цукру в крові частіше знаходиться в цільовому діапазоні. У більшості людей часто буває менше підйомів і спадів	Вам потрібно буде постійно носити помпу з собою. Знімайте її лише на невеликі перерви, наприклад, коли ви плаваєте або приймаєте душ
Вам не доведеться робити ін'єкції так часто	Інфузійний набір (трубки) іноді може засмічуватися, тому вам може знадобитися його швидка заміна
Вам буде легше підбирати дозу інсуліну до, під час і після фізичних навантажень	Вам буде потрібно багато часу, щоб вивчити роботу помпи, особливо, коли ви вперше її отримуєте
Ви матимете більше свободи у виборі того, що, коли і скільки ви їсте	Завжди існує невеликий ризик інфікування через канюлю
Підвищиться точність при зниженні високого рівня цукру	Іноді вам все одно доведеться робити ін'єкції

1.7.2 Інсулінова ручка

Можна вводити інсулін за допомогою інсулінової шприц-ручки та голки, або деякі люди використовують інсулінові флакони та шприци замість шприц-ручки.

Існує два типи інсулінових шприц-ручок:

- шприц-ручка, в якій вже є інсулін, і яку потрібно викинути після того,

як вона спорожніє;

- шприц-ручка, яку можна використовувати повторно з картриджами з інсуліном, які ви міняєте;
- Smart інсулінові ручки.

Розумні інсулінові ручки можуть передавати інформацію про ваші дози інсуліну на мобільний додаток. Такі ручки також можуть автоматично записувати деталі про прийом інсуліну. Для користування розумною ручкою потрібен смартфон або планшет з технологією ближнього радіозв'язку (NFC).

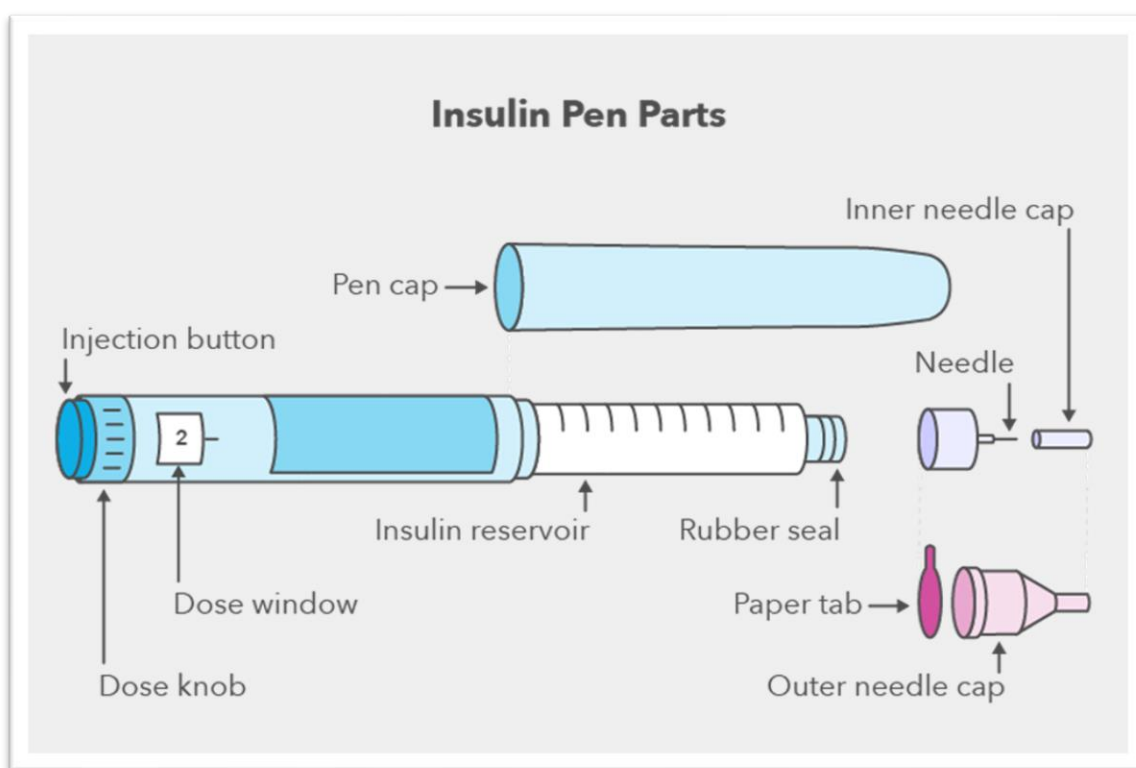


Рисунок 1.9 – Інсулінова шприц-ручка

1.8 Способи вимірювання рівня цукру в крові

1.8.1 Безперервні глюкометри

Глюкометри з флеш-датчиком і безперервні глюкометри дозволяють перевіряти рівень цукру без необхідності колоти пальці. Носіння невеликого датчика, який зчитує рівень цукру в крові протягом дня і ночі, дає можливість

отримувати інформацію на мобільний телефон або інший пристрій. Якщо хтось стежить за діабетом, його мобільний телефон також може бути підключений до цього датчика.

CGM та глюкометри, такі як Freestyle Libre 2, дають можливість налаштувати будильник, який спрацює, якщо рівень цукру в крові буде надто низьким або високим. Сигнал може бути також налаштований на мобільний телефон іншої особи, наприклад, батьків або опікуна.

Одна з головних переваг глюкометрів та CGM - можливість переглядати зміни рівня цукру щохвилини протягом дня і ночі. За допомогою діаграм і графіків можна бачити, як їжа, активність та інші фактори впливають на рівень цукру в крові.

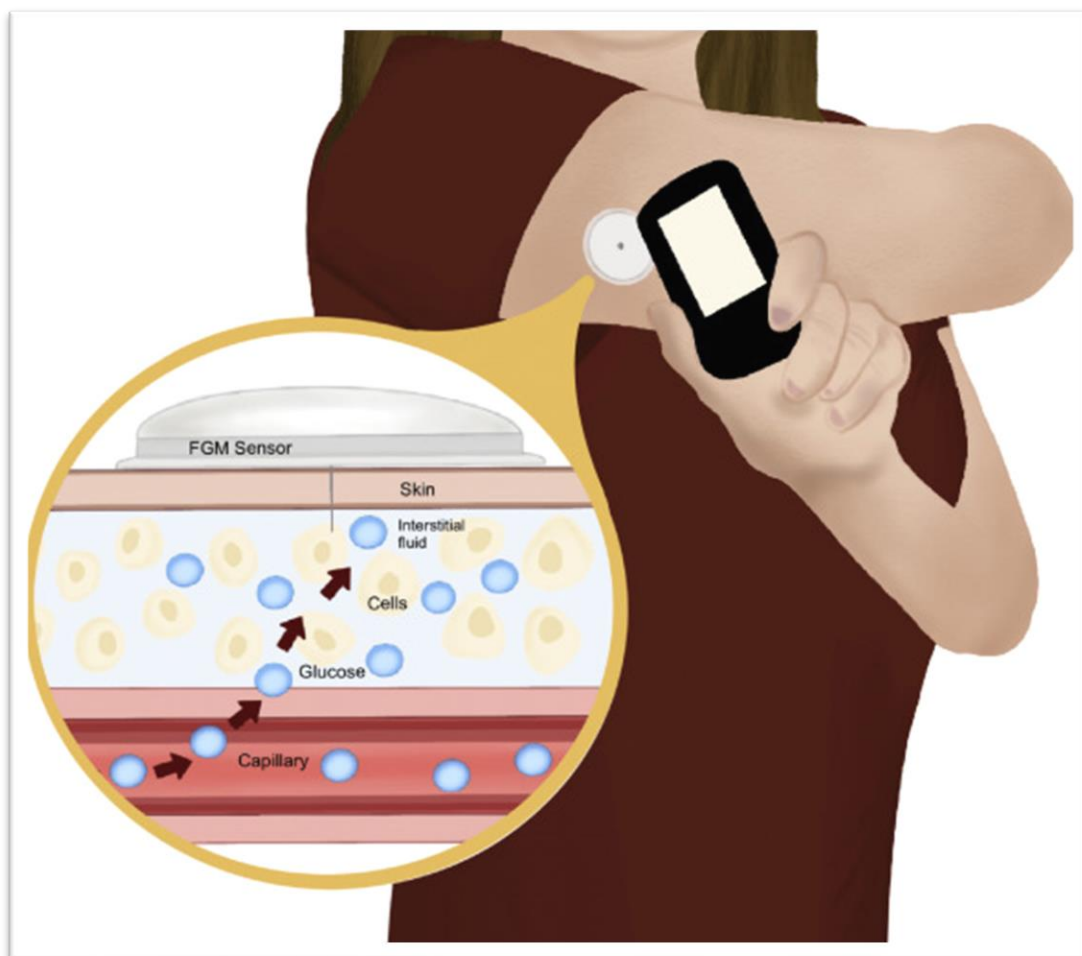


Рисунок 1.9 Глюкометр з флеш-датчиком

1.8.2 Тест-смужки та глюкометри

Тест-смужки та глюкометри - це традиційний спосіб контролю рівня цукру в крові. Ви використовуєте тест-смужки, щоб швидко визначити рівень цукру в крові та контролювати його діапазон.

Глюкометр - це засіб перевірки рівня цукру в крові, який необхідний при лікуванні діабету інсуліном. Стандартний глюкометр має ланцет для проколювання пальця, цифровий дисплей і місце для вставки тест-смужки.

Можна знайти різноманітні типи глюкометрів з додатковими функціями, наприклад, з USB-роз'ємом, що дозволяє зберігати показники на комп'ютері. Також існують глюкометри з калькулятором для розрахунку співвідношення інсуліну до вуглеводів, які можна підключити до смартфона.

Також можна придбати безсмужковий глюкометр, де тест-смужки вже встановлені, і не потрібно вставляти їх окремо. У такому глюкометрі смужки обертаються в приладі замість того, щоб їх замінювати кожного разу. Якщо касета закінчується, треба замінити її, і зазвичай у касеті є 50 смужок.

Для визначення рівня цукру в крові тест-смужка вставляється в глюкометр, якщо вона ще не встановлена. Після проколу пальця глюкометр дозволяє прикласти тест-смужку до краплі крові. Це дозволяє отримати результат вимірювання рівня цукру в крові з глюкометра.

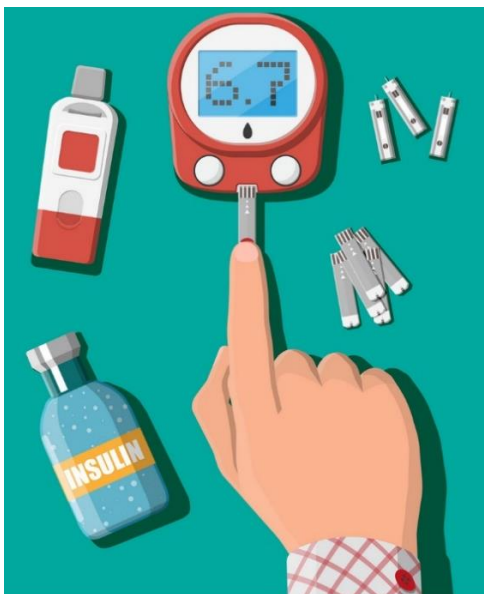


Рисунок 1.10 - Тест-смужки та глюкометри

1.9 Аналіз та огляд аналогів

1.9.1 Beetis

Beetis - це компанія, яка розробляє програмне забезпечення для керування діабетом. Їхній додаток пропонує рішення для моніторингу рівня цукру в крові, керування прийомом ліків і харчуванням, а також співпрацю зі спеціалістами зі здоров'я для покращення управління цим захворюванням.

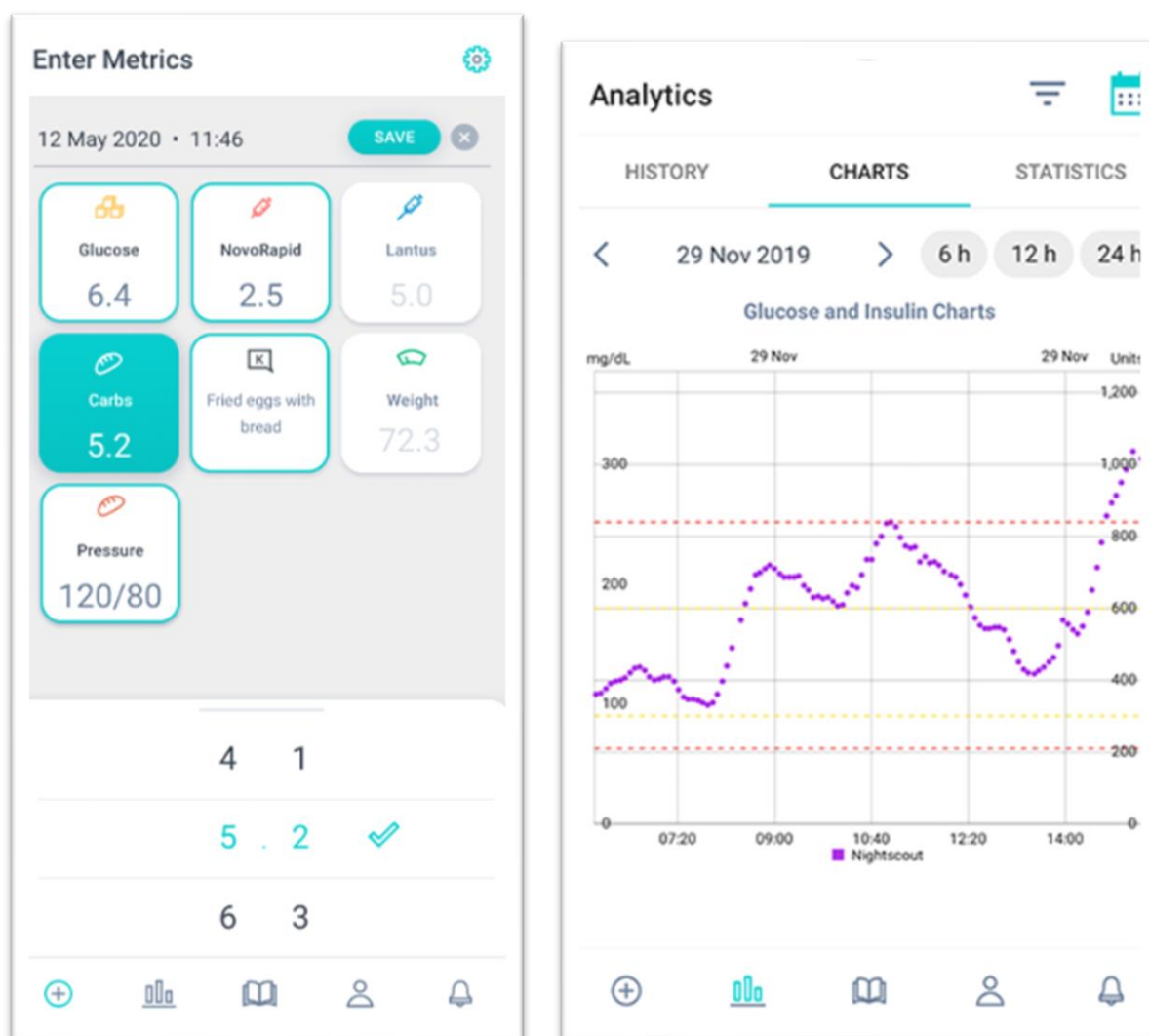


Рисунок 1.11 Скріншоти додатку Beetis

Згідно з їхнім веб-сайтом, компанія має на меті зробити керування діабетом більш ефективним та зручним для людей, які мають це захворювання. Beetis надає своїм користувачам можливість моніторити свій стан здоров'я, спілкуватися зі спеціалістами зі здоров'я та використовувати інші інструменти для кращого

керування діабетом. Окрім того, компанія заявляє, що використовує передові технології, такі як штучний інтелект, для підтримки користувачів у веденні здорового способу життя.

Але в цього додатка є кілька недоліків, по-перше незручно ним користуватись, так як інтерфейс досить складний в розумінні, по-друге калькулятор для розрахунку дози інсуліну доступне тільки по платній підписці, хоча і є пробний період у 183 дні, але коли час вийде, ви повинні сплатити, або не користуватись калькулятором.

Plan Name	Price	For	Key Features	Action
Observation	€0 /mo	observer	<ul style="list-style-type: none"> Observation feature Food Database No ads Up to 10 users Enter measurements * Carbs Calculator * Bolus Wizard * Nightscout Data * 	Download
"Monthly" plan	€1,49 /mo	user	<ul style="list-style-type: none"> Observation feature Food Database No ads Up to 10 observers Enter measurements * Carbs Calculator Bolus Wizard Nightscout Data 	Subscribe

* - when observing user with subscription

Free trial 183 days

Рисунок 1.12 Тарифні плани Beetis

Також незручний у користуванні калькулятор, щоб ввести значення треба нажимати на стрілочку щоб змінювалась цифра, немає можливості вводити клавіатурою.

Додаток Veetis доступний на платформі Android в Google Play та має оцінку 5,0 з 135 000 відгуків. Це завдяки таким його перевагам:

- Зручний спосіб вести щоденник з введених доз;
- Можливість додати декілька користувачів(наприклад, батьків);
- Налаштування повідомлення(наприклад, коли треба їсти);
- Зберігання даних на акаунті(допоможе не втратити всі показники при втраті телефону).

1.9.2 DiabTrend

DiabTrend лідирує в рейтингу додатків для діабетиків, оскільки він надає безкоштовно функції, які інші додатки пропонують тільки за плату. DiabTrend ставить користувачів з діабетом на перше місце, надаючи їм засоби самоконтролю і допомагаючи зробити значні зміни у способі життя, які необхідні для керування діабетом 1-го типу та покращення стану у випадку діабету 2-го типу.

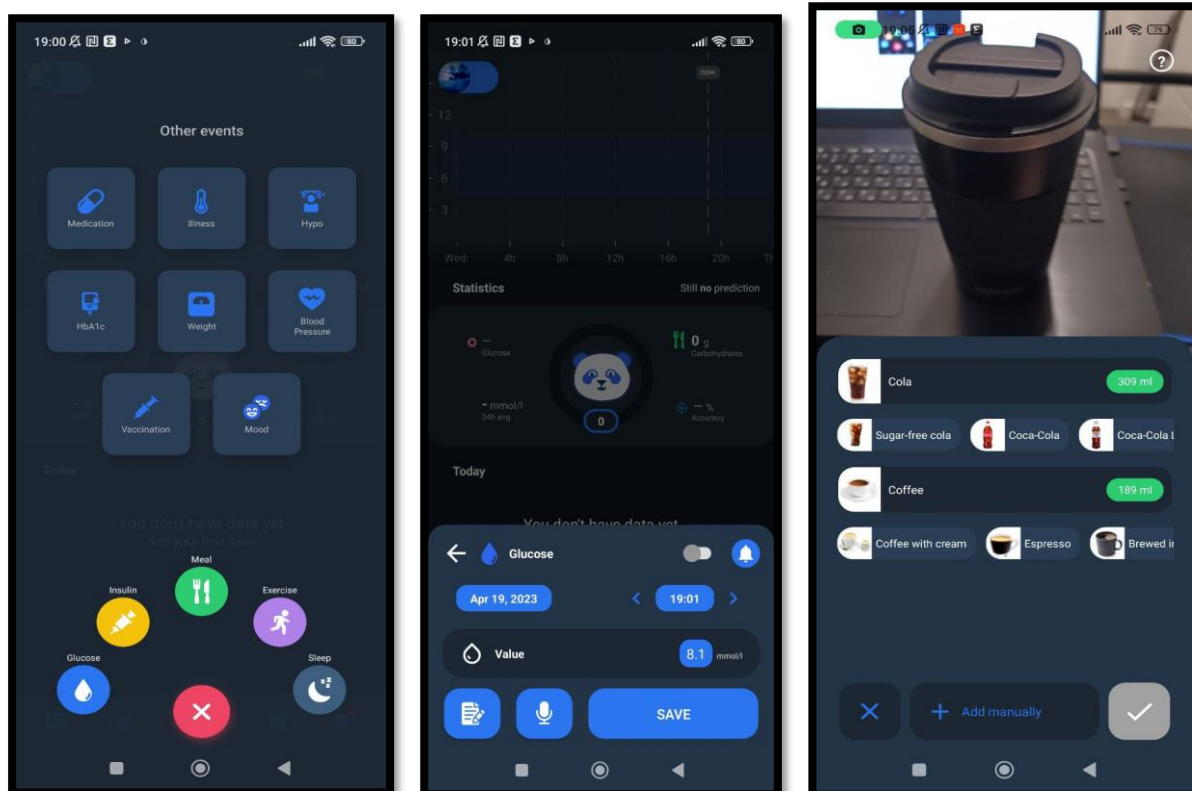


Рисунок 1.13 Скріншоти додатку DiabTrend

Цей додаток є інноваційним інструментом для ведення щоденника діабету, який використовує штучний інтелект. Один з його основних функцій - відстеження їжі за допомогою камери та швидкого розпізнавання продуктів:

1. Автоматичний підрахунок калорій і вуглеводів.

2. Вбудований штучний інтелект навчається на основі вашого щоденника діабету, щоб допомогти вам приймати кращі рішення. Дозволяє прогнозувати рівень глюкози в крові до 4 годин, допомагаючи вам уникнути гіпо- та гіперглікемії.

3. Інтеграція з датчиками та додатками для здоров'я, такими як Apple Fit та Google Fit для моніторингу.

4. DiabTrend орієнтований на пацієнта, що дозволяє вам, пацієнту, ділитися даними та візуалізацією вашого стану з вашим поточним лікарем. Зміна місця проживання або лікаря не повинна означати, що вам потрібно змінювати додаток, який ви використовуєте, забезпечуючи безперервність даних.

5. Професійна звітність про стан здоров'я.

6. працює в режимі офлайн, що дозволяє уникнути незручностей, пов'язаних з додаванням даних через перебої в роботі інтернету.

7. Добре зарекомендував себе, активний розробник, який реагує на відгуки та пропозиції щодо покращення функцій, надаючи при цьому доброзичливу та чуйну технічну підтримку.

8. Впроваджує інновації, коли відкриваються нові фактори ризику діабету, щоб покращити лікування.

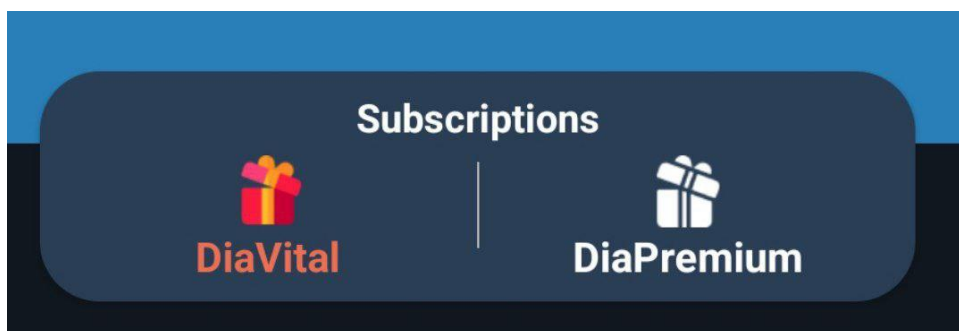


Рисунок 1.14 Підписки додатку DiabTrend

DiaVital - базовий план додатку, який доступний безкоштовно, і містить велику кількість функцій, включаючи можливість створювати звіти. Однак, за плату, користувач може отримати доступ до розширених функцій, які включають розпізнавання їжі на основі штучного інтелекту, прогнозування рівня глюкози в крові, глікемічного індексу та бази даних рецептів для продуктів харчування. Цей платний план називається DiaPremium.

Недоліки:

- Відсутність калькулятора дози інсуліну в залежності від вуглеводів
- Обмежена кількість мов
- Немає синхронізації з CGM(Continuous Glucose Monitoring)

1.9.3 mySugr

mySugr є додатком з великою кількістю користувачів у Німеччині, де він навіть рекомендується лікарями в рамках національної системи охорони здоров'я. Додаток може синхронізуватися з глюкометрами серії Accu-Chek відносної легкої, має багато корисних функцій, серед яких особливо варто відзначити:



Рисунок 1.15 Додаток mySugr та глюкометр Accu-Chek

- mySugr Pump Control, нещодавнє доповнення, що дозволяє користувачам помп для інсуліну вводити болюси через додаток mySugr на платформі Android. Крім того, він містить калькулятор болюсів, що дозволяє легко визначити оптимальну дозу інсуліну для кожного прийому їжі;
- Додатковою можливістю для більш особистої настройки є налаштований екран входу в систему, який розташовує потрібні функції саме там, де їх шукає користувач, що підвищує зручність використання додатку.

Великим плюсом цього додатку є наявність болюсного калькулятора (калькулятор дози інсуліну в залежності від спожитої їжі) він простий в користуванні та є дуже ефективним для покращення життя інсулінозалежних людей.

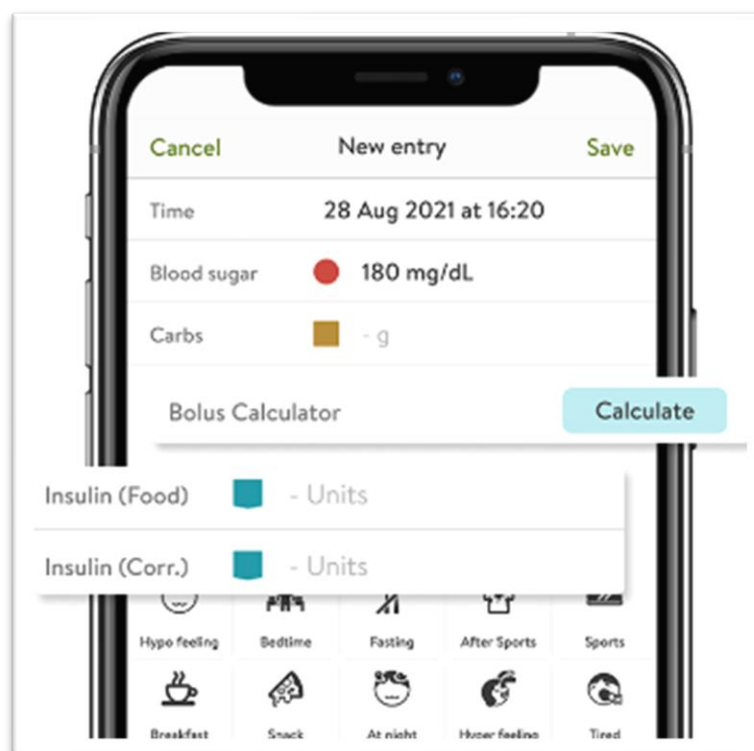


Рисунок 1.16 Болюсний калькулятор додатку mySugr

Але є суттєвий недолік. Цей додаток недоступний в Україні.

Таблиця 1.3 – Порівняння аналогів

	Beetis	DiabTrend	mySugr
Платформи	Android	Android, iOS, Web	Android, iOS
Наявність української мови	-	-	-
Болюсний калькулятор в безкоштовній версії	-	-	+
Болюсний калькулятор в Платній версії	+	-	+
Доступне в Україні	+/-	+	-
Синхронізація з CGM	+	-	-
Можливість порахувати дозу інсуліну	-	-	-

1.10 Аналіз дослідження та встановлення вимог до ПЗ

Проаналізувавши плюси та мінуси додатків-аналогів по поліпшенню життя інсулінозалежним людям можна зробити висновок:

- В цих додатках немає доступного, зручного та простого калькулятора дози інсуліну в залежності від спожитої їжі;
- Також всі додатки зроблені на всі платформи окрім Desktop.

При створенні програмного забезпечення будуть враховані більшість переваг та недоліків існуючих додатків.

2. ОГЛЯД ЗАСОБІВ СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Вибір середовища програмування

2.1.1 Visual Studio Code

Visual Studio Code, також відомий як VS Code, є редактором вихідного коду, що був розроблений Microsoft за допомогою Electron Framework для операційних систем Windows, Linux та macOS. Серед ключових функцій VS Code є підтримка налагодження, підсвічування синтаксису, інтелектуальне завершення коду, фрагменти, рефакторинг коду та вбудований Git. Користувачі можуть налаштовувати тему, комбінації клавіш, налаштування та встановлювати розширення для додавання нових функцій.

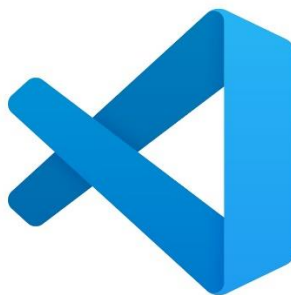


Рисунок 2.1 Логотип Visual Studio Code

У опитуванні Stack Overflow 2022, що брало участь 71 010 респондентів, Visual Studio Code став найбільш популярним інструментом розробки, з 74,48% відповідей, що підтверджують використання цього редактора коду.

Visual Studio Code - це редактор вихідного коду, який може бути використаний для програмування на різних мовах, таких як C, C#, C++, Fortran, Go, Java, JavaScript, Node.js, Python та Rust. Він заснований на фреймворку Electron, який використовується для створення веб-додатків на Node.js, який працює на движку верстки Blink. Visual Studio Code використовує той самий редактор компонентів (під кодовою назвою "Monaco"), що використовується в Azure DevOps (раніше відомий як Visual Studio Online та Visual Studio Team Services).

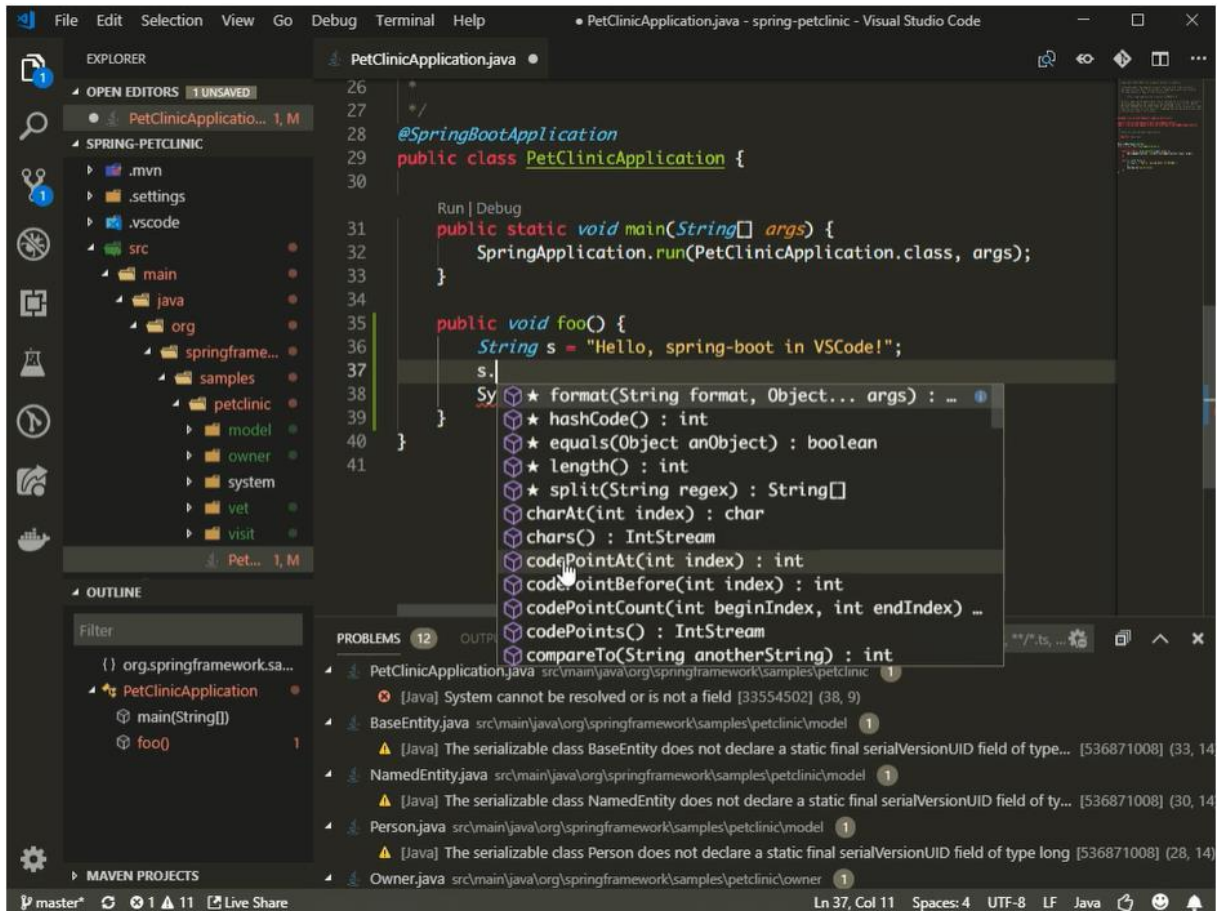


Рисунок 2.2 Інтерфейс Visual Studio Code

Visual Studio Code забезпечує базові функції Java, включаючи автодоповнення коду, перепакунання, лінтування, форматування та фрагменти коду, а також підтримку налагодження та модульного тестування. Він інтегрується з різними інструментами та фреймворками, такими як Maven, Tomcat, Jetty та Spring Boot. За допомогою Visual Studio Code Java розробники можуть ефективно редагувати код та проводити повний цикл розробки, тестування та налагодження.

2.1.2 IntelliJ IDEA

IntelliJ IDEA - це провідне інтегроване середовище розробки, що написано на мові Java, для мов програмування Java та Kotlin. Воно має набір функцій, які допомагають підвищити продуктивність розробника, такі як інтелектуальна підтримка кодування, надійні засоби рефакторингу, миттєва навігація по коду,

вбудовані інструменти розробника, підтримка веб- та корпоративної розробки тощо.



Рисунок 2.3 Логотип компанії Jet Brains

Кожен аспект IntelliJ IDEA створений для того, щоб забезпечити бездоганну роботу з нуля. Вона забезпечує швидкий доступ до всіх функцій та інтегрованих інструментів, важливих для вашої роботи, а також широкий спектр можливостей налаштування. Можна точно налаштувати все для підтримки вашого робочого процесу: встановити ярлики, встановити плагіни, налаштувати інтерфейс на свій смак і багато іншого.

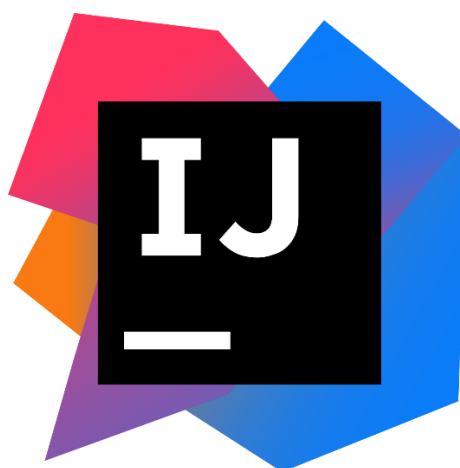


Рисунок 2.4 Логотип IntelliJ IDEA

В IntelliJ IDEA ви можете швидко виконувати різноманітні завдання за допомогою комбінацій клавіш, включаючи перегляд останніх файлів та налагодження вашого проекту. Наприклад, комбінація клавіш подвійного

натискання Shift (Search Everywhere) є універсальним засобом пошуку будь-якого елемента в межах або поза проектом. Пошук може включати файлів, дій, класів, символів, налаштувань, елементів інтерфейсу, а також історію Git.

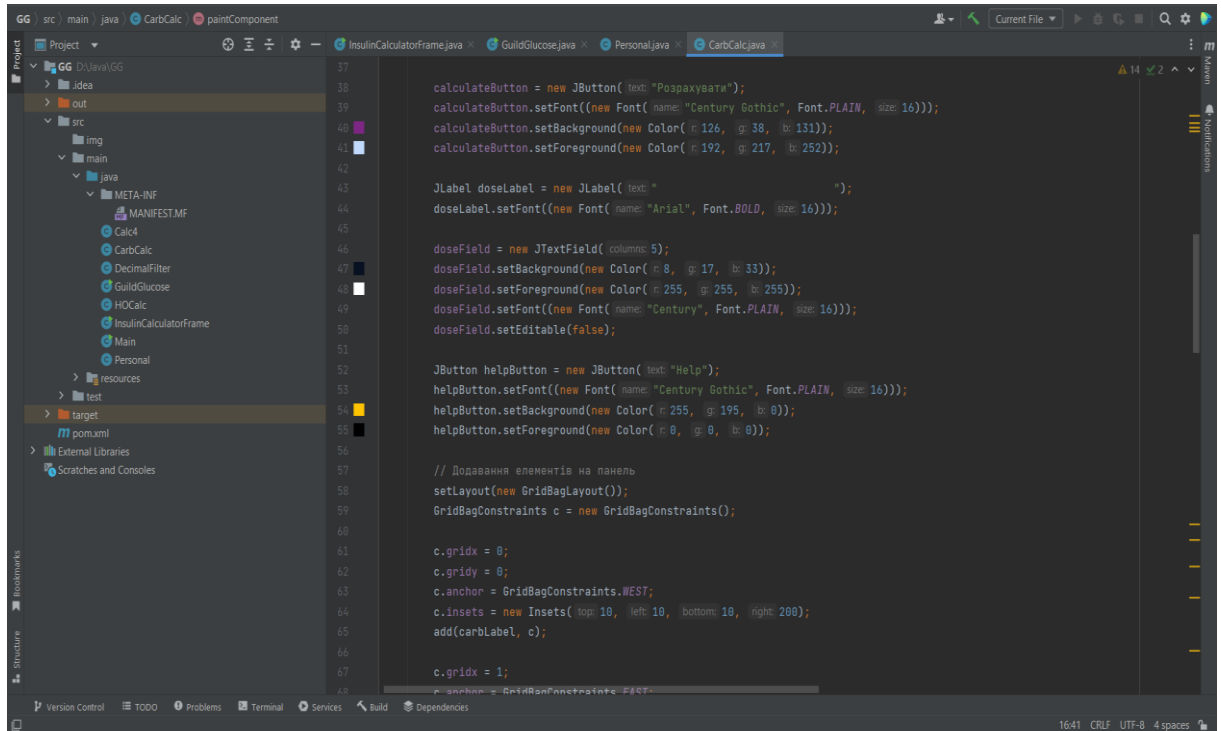


Рисунок 2.5 Інтерфейс середовища програмування IntelliJ IDEA

IntelliJ IDEA має контекстно-залежну функцію автодоповнення, яка пропонує лише ті варіанти завершення коду, що відповідають поточній позиції маркера. Користувачам не потрібно знати жодних спеціальних комбінацій клавіш або налаштовувати параметри - автодоповнення коду починається автоматично в момент введення користувачем коду в редакторі.

2.1.3 NetBeans

NetBeans - це середовище розробки (IDE), яке було створене для розробки додатків на Java, і воно підтримує розробку додатків з модульних програмних компонентів, відомих як модулі. Ця платформа може бути використана на різних операційних системах, таких як Windows, macOS, Linux та Solaris. NetBeans також має підтримку розширень для інших мов програмування, таких як PHP, C, C++,

HTML5 та JavaScript. Більше того, додатки, створені на основі NetBeans, можуть бути розширені сторонніми розробниками.



Рисунок 2.6 Логотип IDE NetBeans

NetBeans IDE є відкритим середовищем розробки, що підтримує створення всіх видів Java-додатків (Java SE, JavaFX, Java ME, веб-, EJB та мобільні додатки). У ньому присутні такі функції, як система збирання проектів на основі Ant, підтримка Maven, засоби рефакторингу, можливість контролю версій (з підтримкою CVS, Subversion, Git, Mercurial та Clearcase) та інші корисні функції.

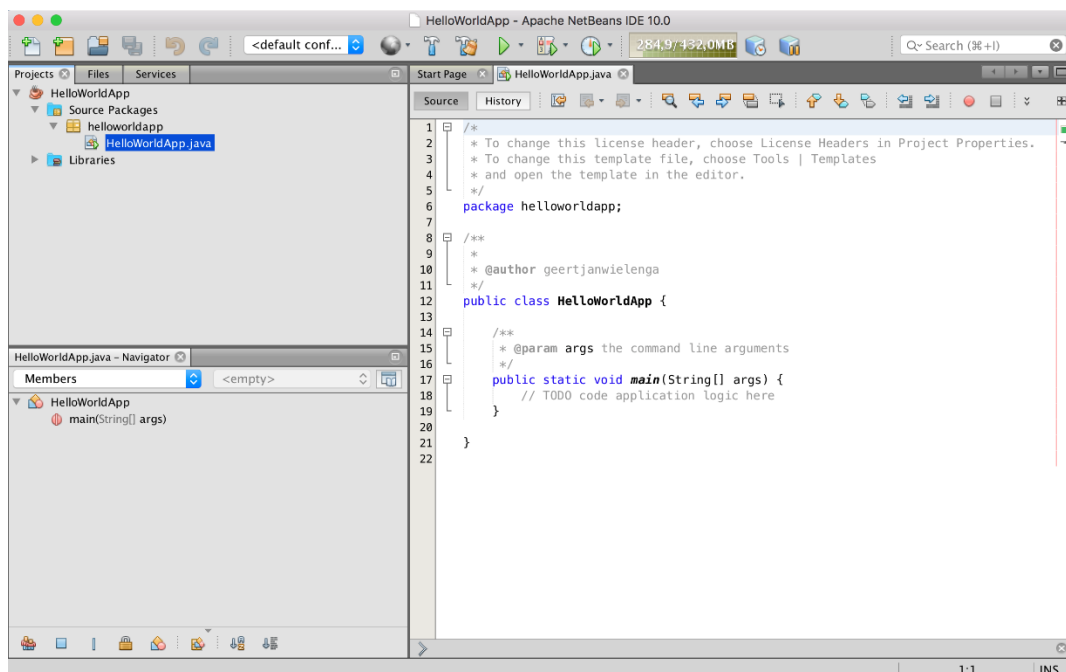


Рисунок 2.7 Інтерфейс IDE NetBeans

NetBeans IDE пропонує модульну систему, де кожен модуль забезпечує конкретну функцію, таку як підтримка мови програмування Java, систем контролю версій або редагування. Установка NetBeans містить всі необхідні модулі для розробки на Java, забезпечуючи швидкий початок роботи. За допомогою модульної системи можна також додавати додаткові функції, наприклад, підтримку інших мов програмування. NetBeans є базою для таких IDE, як Sun Studio, Sun Java Studio Enterprise та Sun Java Studio Creator від Sun Microsystems.

2.2 Вибір бібліотеки для GUI

2.2.1 AWT

Abstract Window Toolkit - це платформозалежний інструментарій Java для створення вікон, графіки та віджетів користувацького інтерфейсу, який був створений до Swing. Він є частиною Java Foundation Classes - стандартного API для надання GUI для програм на Java, а також інструментарієм графічного інтерфейсу для деяких профілів Java ME. Наприклад, деякі профілі для мобільних пристроїв вимагають час виконання Java для підтримки інструментарію абстрактних вікон.



Рисунок 2.8 GUI з використанням AWT

Якщо для AWT-компонента існує версія у Swing, то потрібно використовувати саме її, яка починається з букви J, і уникати використання AWT-версії. Наприклад, для створення кнопок у Swing необхідно використовувати клас

JButton, а не клас Button. Але основні класи AWT, такі як Color і Font, можна використовувати у Swing без змін.

При малюванні у Swing використовується JPanel та перевизначається paintComponent(Graphics g) замість використання методів AWT paint().

До Java 6 Update 12, комбінування компонентів Swing та базових віджетів AWT призводило до непередбачуваних наслідків, наприклад, віджети AWT можуть з'являтися над віджетами Swing, незалежно від їх порядку. Ця проблема виникала через значну різницю в архітектурі рендерингу віджетів у двох інструментаріях, хоча Swing взяв у спадок важливі верхні контейнери від AWT.

Починаючи з Java 6 Update 12, можна змішувати віджети Swing та AWT без проблем з порядком.

2.2.2 Swing

Swing - це інструментарій віджетів графічного інтерфейсу, що використовується для створення GUI-програм на Java. Він є частиною Java Foundation Classes (JFC), яке є стандартним API для розробки графічного інтерфейсу користувача в програмах на Java і надає великий набір компонентів, які можна використовувати для створення різноманітних GUI-елементів.

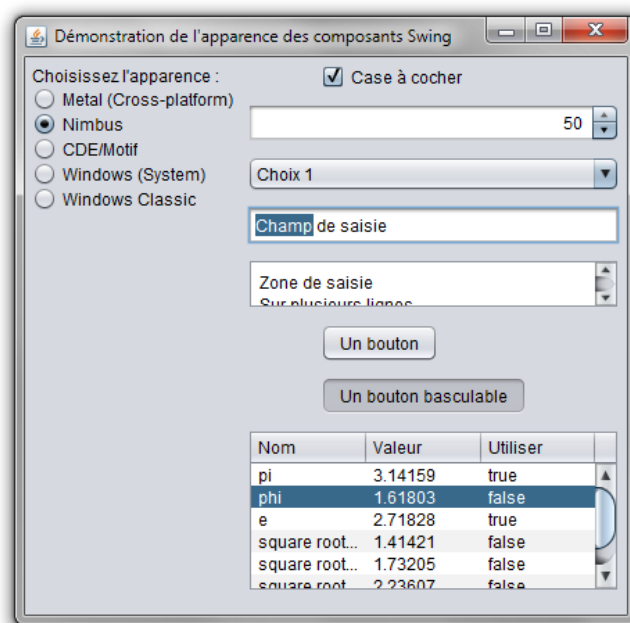


Рисунок 2.9 GUI з використанням Swing

Swing - це фреймворк графічного інтерфейсу для Java, який є незалежним від платформи і використовує модель-вид-контролер. Він працює за однопотоковою моделлю програмування і надає рівень абстракції між структурою коду та графічним представленням графічного інтерфейсу, що базується на Swing. Фреймворк надає великий набір компонентів, які можна використовувати для створення складних GUI-елементів.

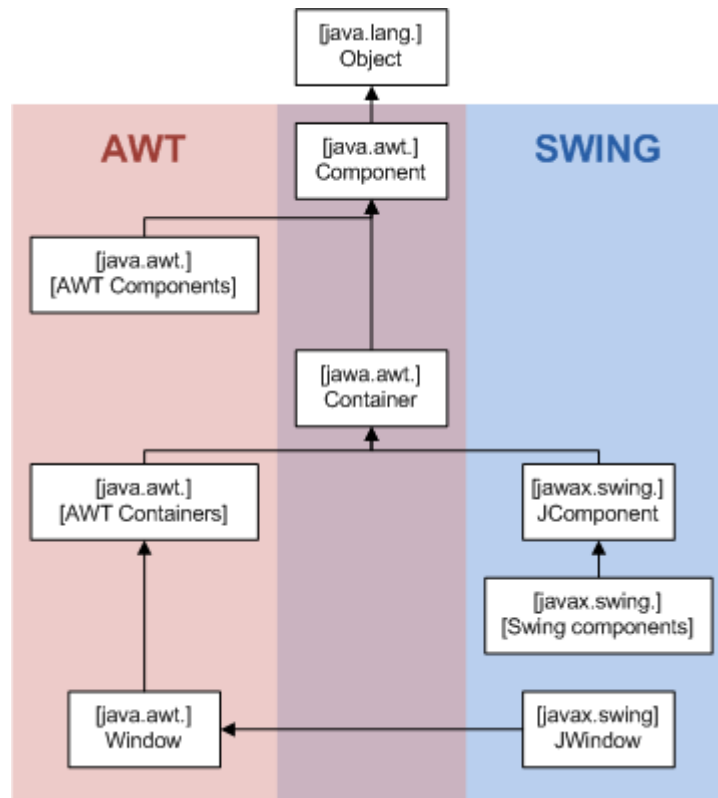


Рисунок 2.10 Ієрархія класів AWT та Swing

Swing є платформонезалежним фреймворком для графічного інтерфейсу, оскільки він був розроблений повністю на Java. Документацію для всіх класів Swing можна знайти в Java API Guide для версії 6 або Java Platform Standard Edition 8 API Specification для версії 8.

2.2.3 JavaFX

JavaFX - це платформа розробки програмного забезпечення, яка надає інструменти для створення десктопних та веб-додатків, що можуть працювати на

різних платформах, включаючи Microsoft Windows, Linux, macOS, а також на мобільних пристроях з операційними системами iOS і Android. JavaFX надає широкі можливості для розробки графічного інтерфейсу користувача та підтримує різні технології, такі як 2D- та 3D-графіка, мультимедіа, анімація та багато іншого.

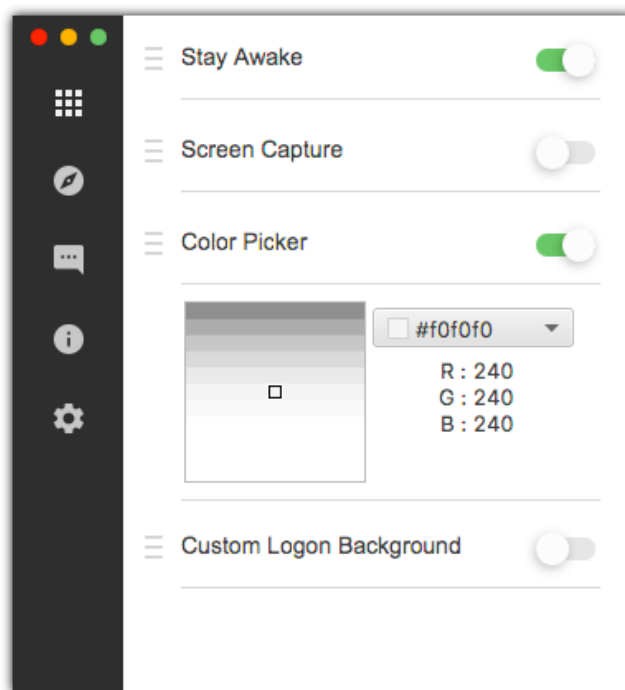


Рисунок 2.11 GUI з використанням JavaFX

JavaFX була запланована як заміна Swing у якості стандартної бібліотеки графічного інтерфейсу для Java SE, але з часом була виключена зі стандартних версій. Swing та AWT залишилися включеними, ймовірно через те, що ринок JavaFX був розмитим зростанням мобільних та веб-додатків. Однак у 2018 році JavaFX була включена до проекту OpenJFX як частина OpenJDK, щоб прискорити її розвиток. Oracle надалі буде підтримувати JavaFX для Java JDK 8 до березня 2025 року.

JavaFX 1.1 базується на концепції "спільного профілю", який може бути використаний на будь-яких пристроях, що підтримують JavaFX. Цей підхід дозволяє розробникам використовувати одну модель програмування для створення додатків, які працюють як на настільних, так і на мобільних пристроях, і взаємодіяти між ними. Крім того, в платформі JavaFX 1.1 є спеціальні API для

конфігурування додатків для конкретних класів пристроїв, включаючи настільні і мобільні. Наприклад, профіль JavaFX Desktop містить Swing і додаткові візуальні ефекти. Такий підхід дозволяє розробникам спільно використовувати код, графічні ресурси і контент між десктопною і мобільною версіями додатків.

2.3 Підсумок та остаточний вибір інструментів

Для написання програмного забезпечення було обране середовище програмування від компанії JetBrains - IntelliJ IDEA. Ця платформа має весь потрібний функціонал в безкоштовній версії, також вона є найбільш адаптованою під Java-проекти. Вона написана мовою Java та має всі необхідні інструменти для реалізації завдання. Також в цій IDE є можливість обирати версії JDK (Java Development Kit) від різних компаній, а також обирати версію на який буде створений проект. Мною був обраний JDK від компанії Oracle OpenJDK версії 18.0.2. Ця версія вийшла на початку 2022 року та є найбільш стабільною на момент написання програми.

Для реалізації GUI в програмному забезпеченні були обрані дві бібліотеки: AWT та Swing. Від їх об'єднання можна отримати дуже зручний інструмент для створення графічного інтерфейсу. Наприклад, при використанні JTabbedPane, ми створюємо програму з вкладками, це створюється за допомогою JFrame який знаходиться в бібліотеці Swing, також всі елементи які ми додаємо на екран будуть створені за допомогою JFrame, а задля їх розміщення ми будемо використовувати GridBagConstraints який міститься в бібліотеці AWT, також додавання кольорів та зміна шрифтів виконується через методи бібліотеки AWT.

3. РЕАЛІЗАЦІЯ ТА ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ СТАБІЛЬНОГО РІВНЯ ЦУКРУ ІНСУЛІНОЗАЛЕЖНИХ ЛЮДЕЙ

3.1 Функціональні вимоги до програмного забезпечення

Таблиця 3.1 – Функціональні вимоги

№	Функціональна вимога
1	Додаток має надавати можливість користувачеві вводити дані про своє харчування, такі як кількість вуглеводів
2	Додаток має надавати можливість користувачеві вводити дані про свій рівень цукру в крові
3	Додаток має розраховувати рекомендовану дозу інсуліну на основі введених даних про харчування та рівня цукру в крові
4	Додаток має мати графічний інтерфейс користувача(GUI)
5	Додаток має зберігати інформацію про вміст вуглеводів у різних продуктах харчування, щоб допомогти користувачеві розраховувати кількість вуглеводів у своїй дієті
6	Додаток має відображати рекомендовану дозу інсуліну в залежності від вмісту вуглеводів у продуктах харчування та введеного рівня цукру в крові користувача

3.2 Планування розробки програмного забезпечення

Після визначення засобів для створення програмного забезпечення треба розбити мету та вимоги на задачі, для успішного створення додатку. Існує багато інструментів для реалізації плану по створенню програмного забезпечення такі як: Jira, Trello, Notion і т.д. Для організації створення додатку було обрано додаток Notion, який має всі необхідні інструменти для цього.

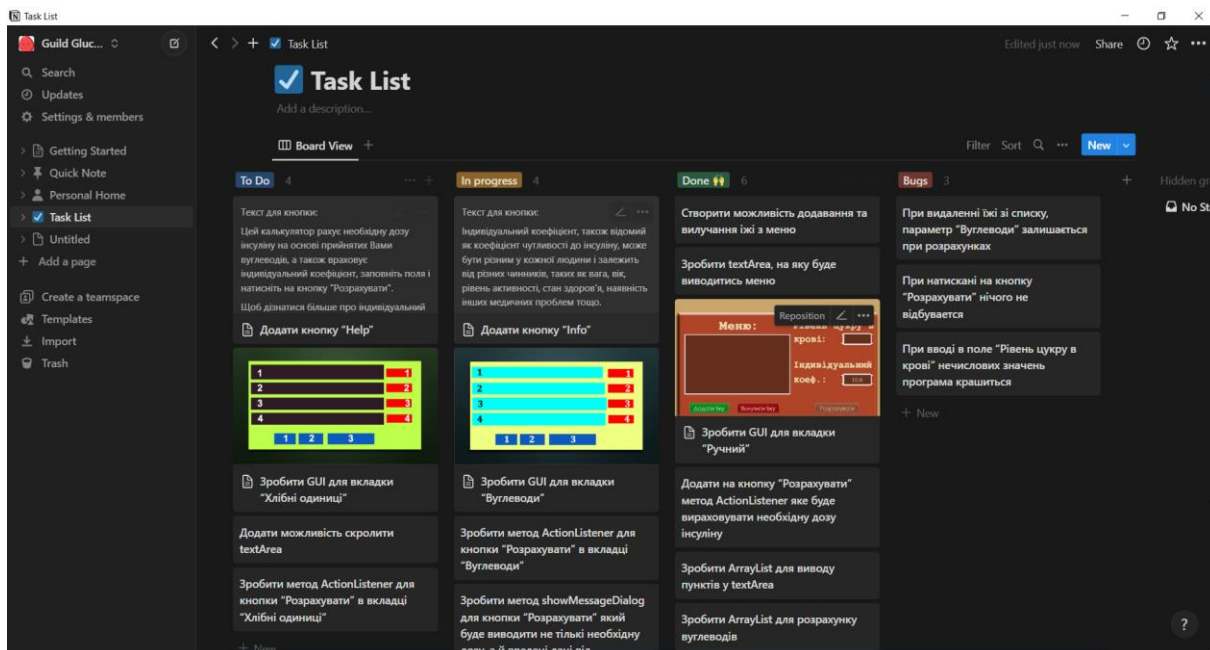


Рисунок 3.1 Task List по створенню додатку в Notion

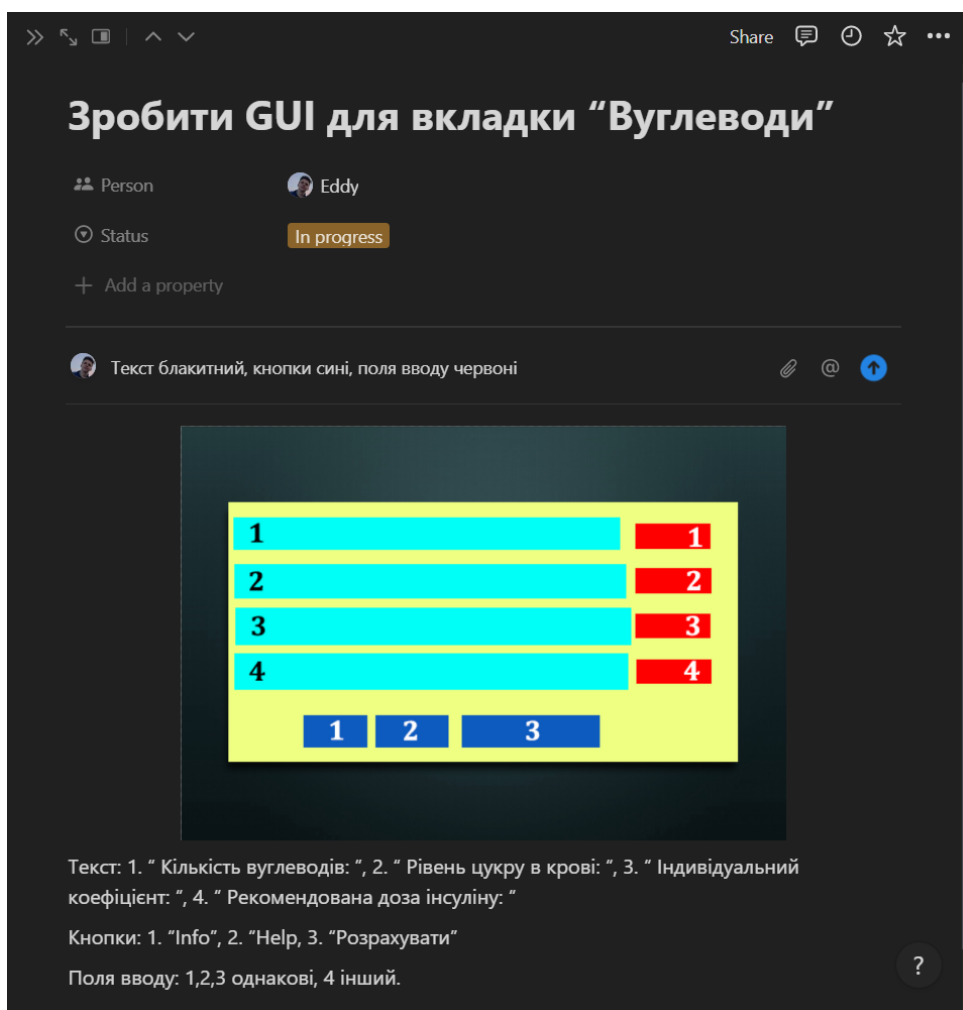


Рисунок 3.2 Task «Зробити GUI для вкладки «Вуглеводи»»

На рисунку 3.1 зображений Task List в якому є пункти «To do», «In progress», «Done» та «Bugs». При створенні необхідної вимоги чи частини програми, я додаю його у список «To do», як тільки я закінчую якусь з цілей у «In progress», я переміщую його туди, після закінчення роботи над цим завданням я його відправляю у колонку «Done». Якщо в процесі розробки, або тестуванні нових виконаних цілей з'являється баг, я додаю новий пункт у «Bugs». Після виправлення, я переміщую цей пункт у архів.

Наприклад візьму завдання «Зробити GUI для вкладки “Вуглеводи”», яка знаходиться у пункті «In progress».

На рисунку 3.2 зображений вигляд однієї з цілей проекту. Додано зображення макету вкладки, яке розробляється. Також на самому зображенні всі елементи різного кольору, щоб можна було зрозуміти де кнопка, а де поле вводу. Також кожен з елементів пронумерований, а під зображенням описано, який елемент за що відповідає, та текст який повинен бути написаний. Після виконання цього завдання я його скопіюю, та додам копію у архів, а сама ціль буде знаходитись у «Done», після цього картинка буде замінена, приклад на рисунку 3.3.

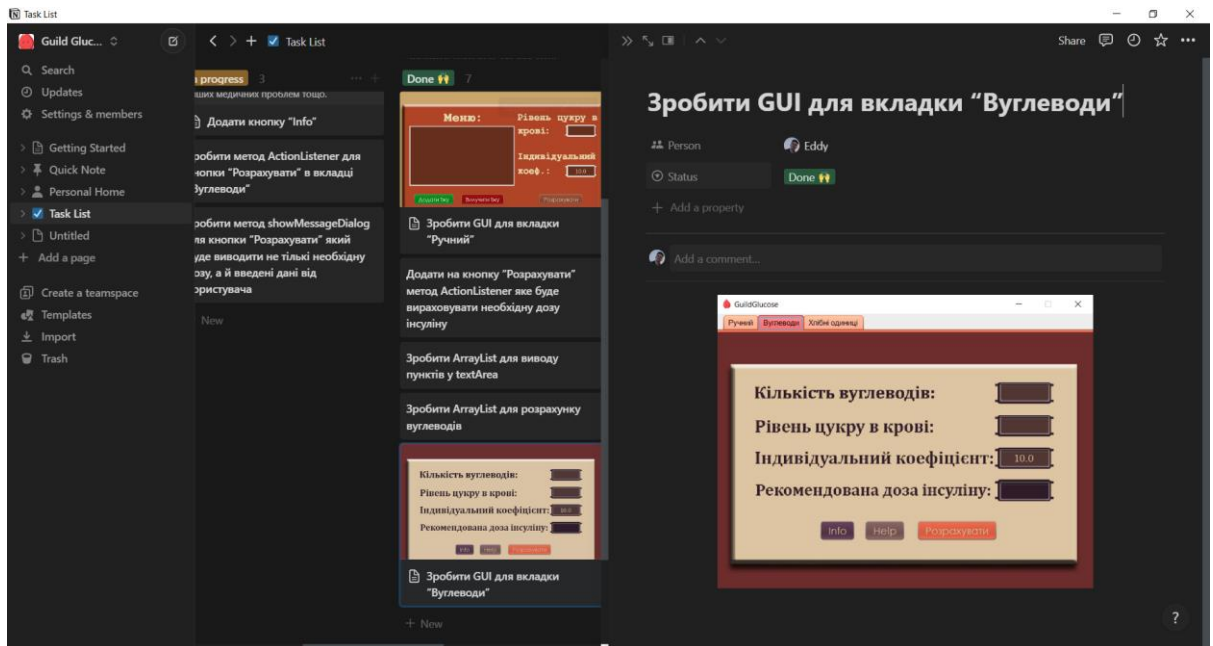


Рисунок 3.3 Done «Зробити GUI для вкладки “Вуглеводи”»

3.3 Визначення функціональних вимог

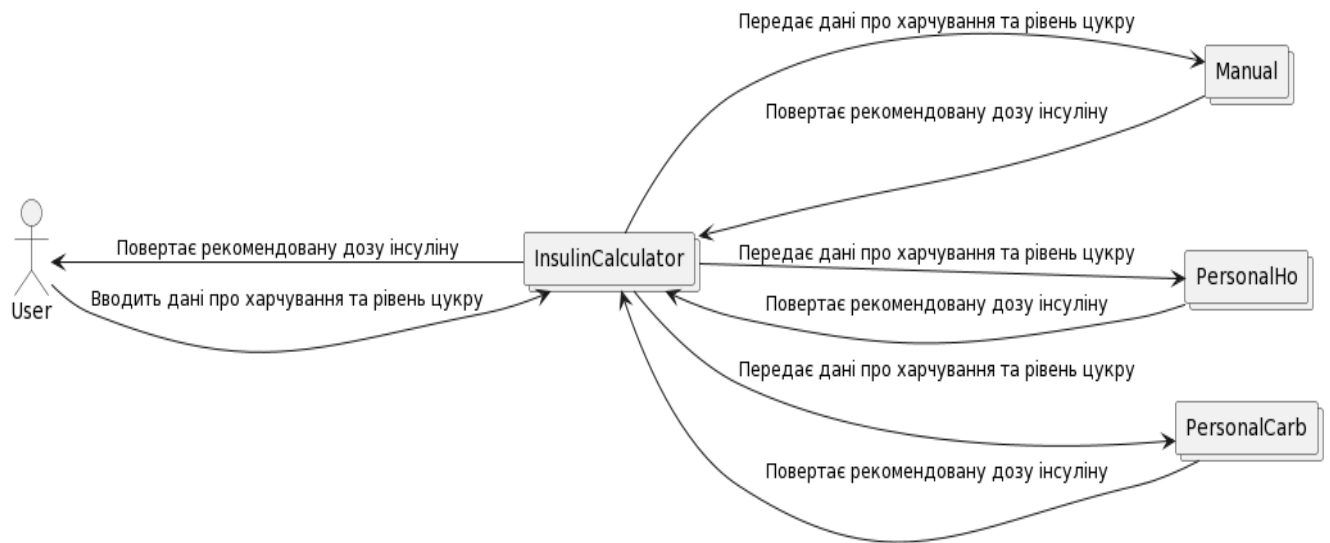


Рисунок 3.4 Use Case-діаграма

У програмному забезпеченні один вид користувача – User. Він передає дані про харчування, рівень цукру в крові та індивідуальний коефіцієнт в клас який є збірником всіх класів GUI – InsulinCalculator. Далі, в залежності від введених даних та обраного засобу цей клас передає далі в один з варіантів підрахунку. Після цього, наприклад, клас Manual отримує від InsulinCalculator дані, які отримав від користувача, та рахує рекомендовану дозу інсуліну. Потім цей клас повертає розрахунки назад в InsulinCalculator, який, в свою чергу, повертає рекомендовану дозу користувачу.

3.4 Алгоритм дій програмного забезпечення

На цій діаграмі(рисунок 3.5) зображені можливі шляхи користувача по програмі.

На початку шляху користувач стикається з вибором способу розрахунку, якщо користувач має вже готове меню, яке порекомендував йому лікар, або він знайшов в інтернеті, та там написана кількість вуглеводів та/або хлібні одиниці, то

користувач обирає шлях який йому потрібен(Розрахунок по вуглеводам або розрахунок по ХО). Це зроблено щоб користувач не вводив наново меню, в якому вже все пораховано. Після вибору користувачеві треба ввести дані про рівень цукру, кількість вуглеводів/ХО та індивідуальний коефіцієнт. Потім він отримає рекомендовану дозу інсуліну

А якщо користувач не має такого меню, він обирає шлях «Розрахунок по їжі», в якому йому потрібно обрати дію додати або вилучити з меню, після внесення свого меню йому потрібно ввести дані про рівень цукру та індивідуальний коефіцієнт. Далі йому буде виведено повідомлення в якому буде вказано необхідну дозу інсуліну щоб покрити їжу яку користувач ввів, меню яке було введено(для перевірки чи все правильно), кількість вуглеводів/ХО в тих пунктах які були внесенні у меню, на випадок якщо користувач хоче зробити цей раціон довгостроковим, щоб у майбутньому він міг користуватись розрахунком по вуглеводам/ХО.

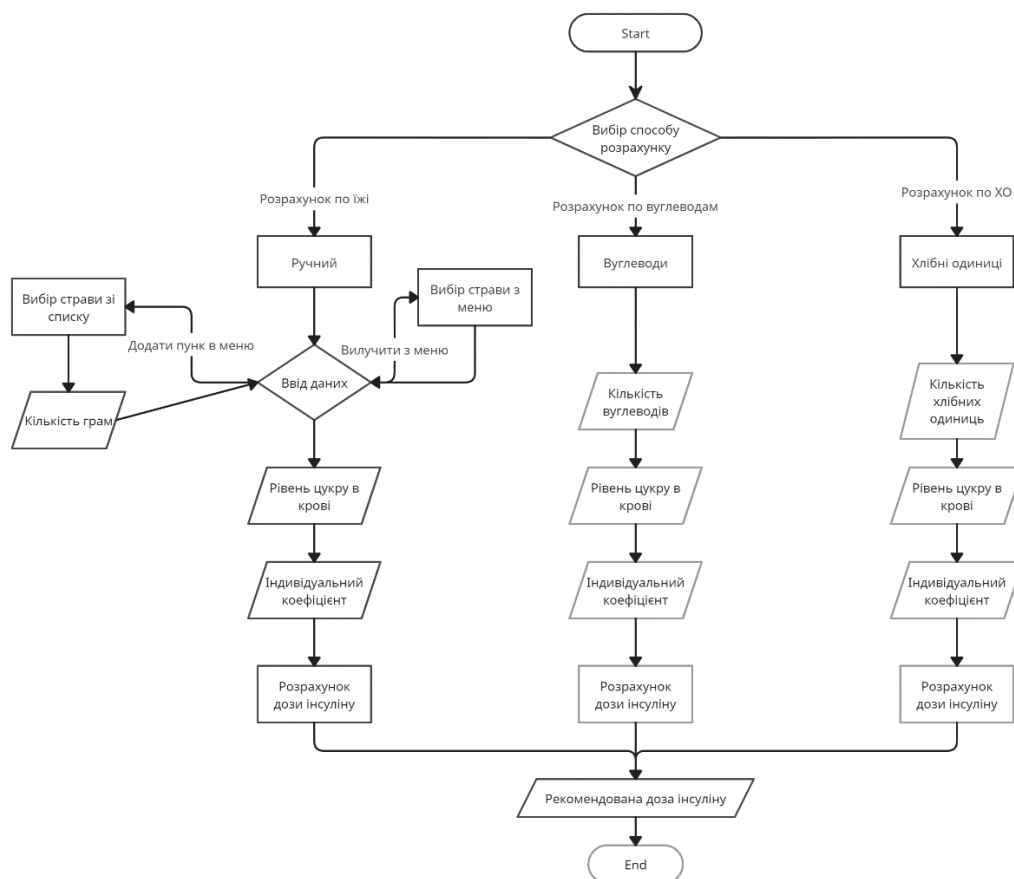


Рисунок 3.5 User Flow діаграма

3.5 Графічний інтерфейс програми

Назва програми –GlucoseGuardian.

Графічний інтерфейс користувача був зроблений з допомогою бібліотек Swing та AWT. А саме бібліотека JFrame, JPanel та JTabbedPane. Інтерфейс був зроблений згідно вимог, та є інтуїтивно зрозумілим та простим у використі. Також для розробки програмного забезпечення був використаний LookAndFeel(LAF) Nimbus.

Термін "Вигляд та Поведінка" (англ. Look and Feel) використовується у розробці програмного забезпечення для опису зовнішнього вигляду та функціональності графічного інтерфейсу користувача (GUI). LAF визначає, як елементи інтерфейсу будуть відображатися на екрані та як вони будуть реагувати на дії користувача. Наприклад, LAF визначає кольори та шрифти для тексту, іконки та курсори, розташування елементів на екрані, а також реакцію на кліки миші та натискання клавіш.

Головним класом виступає InsulinCalculator, який успадковує(extends) такий клас як JFrame. Так як у JFrame є доступ до JComponent, було створено JTabbedPane, завдяки якому було зроблено три вкладки з різними калькуляторами. Також використовувались методи які були успадковані: setDefaultCloseOperation(), setResizable(), setIconImage(), і так далі(всі методи зображені на рисунку 3.6). І так при запуску програми, в додатку є основним клас InsulinCalculator, але так як він сам по собі є контейнером для трьох інших класів, так сказати обгортка, відразу ми бачимо клас Manual.

Першим з функціональних класів є Manual. Він успадковує клас JPanel, який дозволяє додавати на інтерфейс різні елементи, а JFrame є основою для панелей, без додавання JPanel неможливо розташувати на екрані елементи. В цьому класі для зміни фону використовується метод paintComponent(), який міститься у JComponent. Також для створення полів вводу було використано JTextField, цей елемент дозволяє вводити дані, наприклад в моєму ПЗ воно використовується для

вводу рівня цукру в крові та індивідуального коефіцієнту. Для розміщення компонентів на екран використовувався клас `GridBagConstraints` та `GridBagLayout`, які належать бібліотеці AWT.



Рисунок 3.6 Діаграма класів

`GridBagLayout` - це менеджер компоновання в Java, який дає можливість розміщувати компоненти на формі в таблицю з різними розмірами та пропорціями. Він є потужним, але дещо складним у використанні, оскільки кожна клітина таблиці може мати свої власні розміри, пропорції та параметри заповнення. Кожен компонент у таблиці пов'язаний з об'єктом `GridBagConstraints`, який містить параметри розміщення та вигляду компонента. `GridBagLayout` дозволяє дуже точно та гнучко розміщувати компоненти на формі, а правильно налаштовані параметри `GridBagConstraints` допоможуть досягти майже будь-якої розкладки компонентів.

Приклад роботи з цими інструментами на рисунках 3.7 та 3.8.

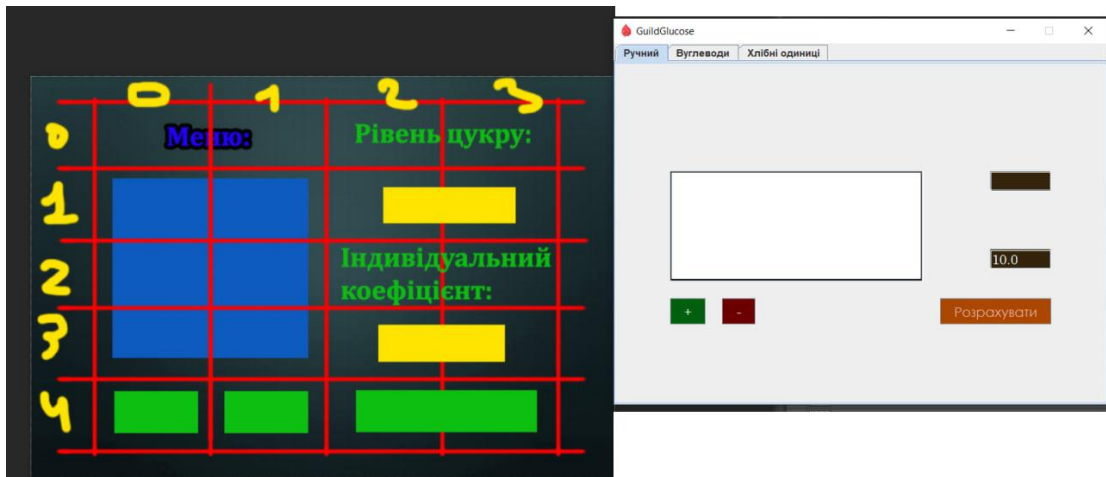


Рисунок 3.7 Приклад розміщення елементів за допомогою GridBagConstraints

На рисунку 3.7 зображено яким чином працює Grid-сітка. У GridBagConstraints є можливість міняти показники `gridx`, `gridy`, `gridwidth`, `gridheight`, `anchor`. Де `gridx` розміщення елемента по горизонталі, `gridy` по вертикалі. На рисунку 3.7 можна наприклад взяти кнопку «+», щоб помістити її у лівий нижній кут треба задати змїну `gridy=4`, а `gridx=0`. Параметри `gridwidth` та `gridheight`, визначають на скільки клітинок елемент буде в висоту або ширину. Наприклад можна взяти текстове поле де повинно відобразитись меню його параметри будуть такі: `gridy=1`; `gridx=0`; `gridwidth=2`; `gridheight=3`;

```
setLayout(new GridBagLayout());
GridBagConstraints c = new GridBagConstraints();

c.gridx = 0;
c.gridy = 0;
c.anchor = GridBagConstraints.CENTER;
c.insets = new Insets( top: 10, left: 10, bottom: 10, right: 10);
c.gridwidth = 2;
add(menuLabel, c);

c.gridx = 0;
c.gridy = 1;
c.anchor = GridBagConstraints.CENTER;
c.insets = new Insets( top: 10, left: 10, bottom: 10, right: 150);
c.gridwidth = 2;
c.gridheight = 3;
add(scrollPane, c);

c.gridx = 0;
c.gridy = 4;
c.anchor = GridBagConstraints.CENTER;
c.insets = new Insets( top: 10, left: 25, bottom: 10, right: 10);
c.gridwidth = 1;
c.gridheight = 1;
add(plusButton, c);

c.gridx = 1;
c.gridy = 4;
c.anchor = GridBagConstraints.WEST;
c.insets = new Insets( top: 10, left: 10, bottom: 10, right: 100);
c.gridwidth = 1;
c.gridheight = 1;
add(minusButton, c);

c.gridx = 2;
c.gridy = 0;
c.anchor = GridBagConstraints.EAST;
c.insets = new Insets( top: 10, left: 10, bottom: 10, right: 10);
c.gridwidth = 2;
c.gridheight = 1;
add(sugarLabel, c);

c.gridx = 2;
c.gridy = 1;
c.anchor = GridBagConstraints.CENTER;
c.insets = new Insets( top: 10, left: 0, bottom: 10, right: 10);
c.gridwidth = 2;
c.gridheight = 1;
add(sugarField, c);

c.gridx = 2;
c.gridy = 2;
c.anchor = GridBagConstraints.EAST;
c.insets = new Insets( top: 10, left: 10, bottom: 10, right: 10);
c.gridwidth = 2;
c.gridheight = 1;
add(coeffLabel, c);
```

Рисунок 3.8 Код з використанням GridBagConstraints та GridBagLayout

За допомогою JScrollPane була створена можливість листати меню, якщо там буде більше 8 пунктів, або якщо один пункт буде більше за двадцять символів.

Також було використано JButton для створення трьох кнопок: «Розрахувати», «Додати» та «Вилучити», на які потім було додано функціонал.

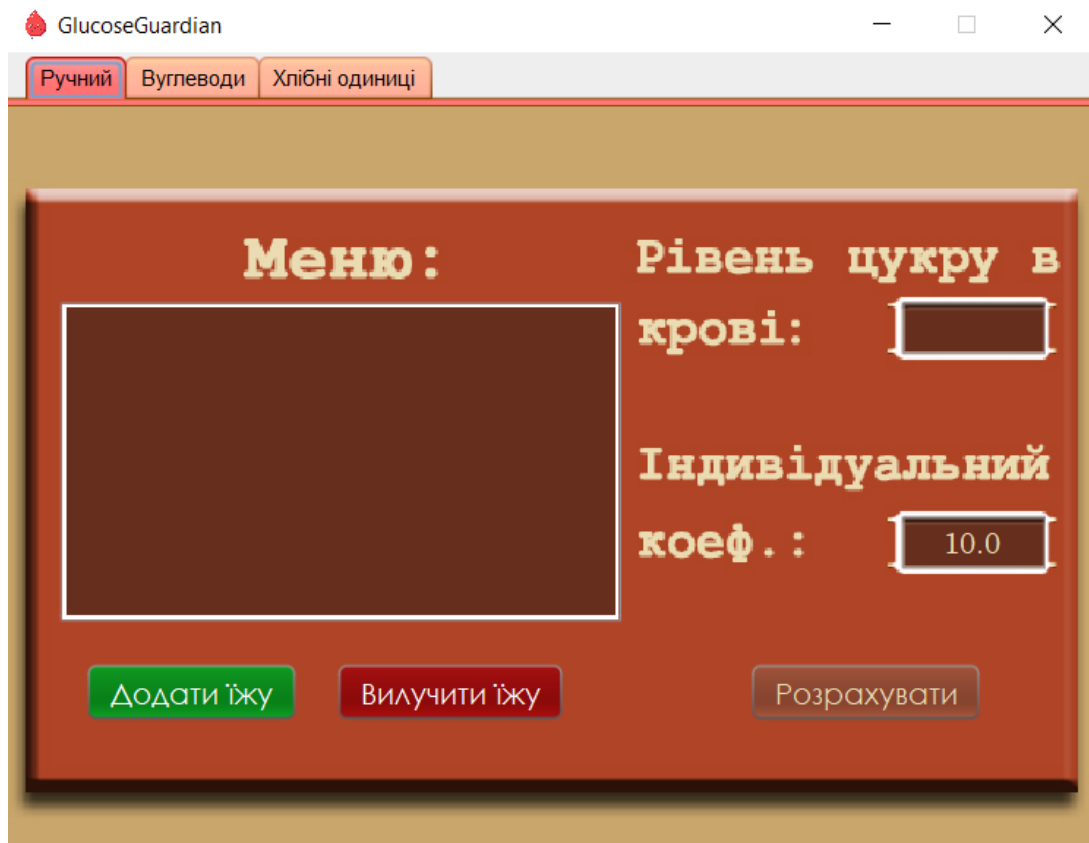


Рисунок 3.9 Вигляд класу Manual

На рисунку 3.9 зображено вигляд програмного забезпечення при запуску. На ньому можна побачити зверху три вкладки, це вибір 3х різних калькуляторів з різним функціоналом. Також на цьому зображенні можна побачити клас Manual.

При натисканні на інші вкладки користувача буде перенесено до інших класів, таких як PersonalCarb та PersonalHo. Вони мають схожий інтерфейс та використовують однакові методи для виводу елементів на екран, але не схожі по функціоналу. Розмітка на JPanel була зроблена також з допомогою GridBagConstraints та GridBagLayout. Основні елементи цих класів це JTextField,

JLabel та JButton. Також використовується `paintComponent()` для розміщення зображення на задній фон.

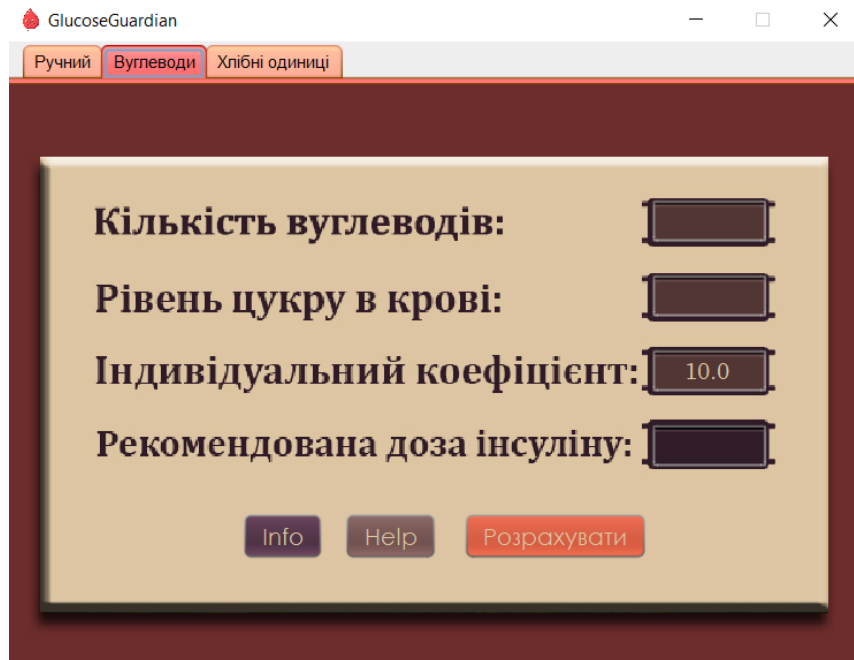


Рисунок 3.10 Вигляд класу PersonalCarb

Також були додані кнопки «Info» та «Help» які інформують користувача як користуватися цим калькулятором.

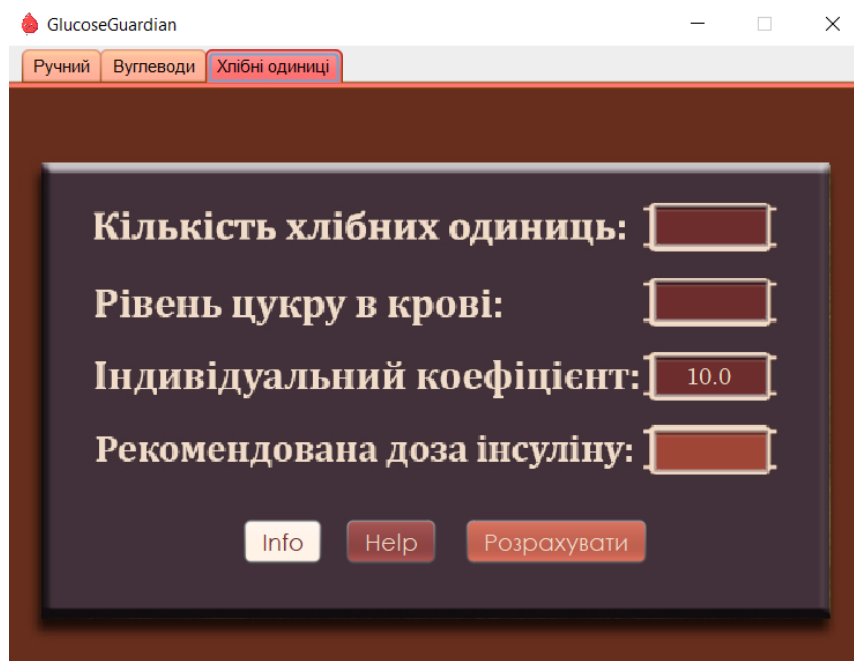


Рисунок 3.11 Вигляд класу PersonalHo

3.6 Функціонал програми

При запуску програми та ініціалізації класу `InsulinCalculator`, створюються об'єкти класів `Manual`, `PersonalCarb` та `PersonalHo`. Після цього в заданих класах відтворюється задана інструкція задана в об'єкті кожного класу. Створення всіх графічних елементів відбувається за межами об'єкту, а в самому порядку вказаних дій відбувається їх зміна та задання кольорів, шрифту, розміру і так далі. Після зміни їх вигляду, через метод `setLayout()` створюється об'єкт класу `GridBagLayout` для розміщення елементів.

Після цього вже відбувається основний функціонал програми. В класі `Manual` є три функціональні кнопки:

- Додати їжу
- Вилучити їжу
- Розрахувати

При натисканні на кнопку «Додати їжу» через метод класу `JButton` – `addActionListener({...})` визивається діалогове вікно через метод класу `JOptionPane`:

```
String n = (String)JOptionPane.showInputDialog(null, "Оберіть їжу зі списку",
"Додати", JOptionPane.QUESTION_MESSAGE, null, options, options[0]);
```

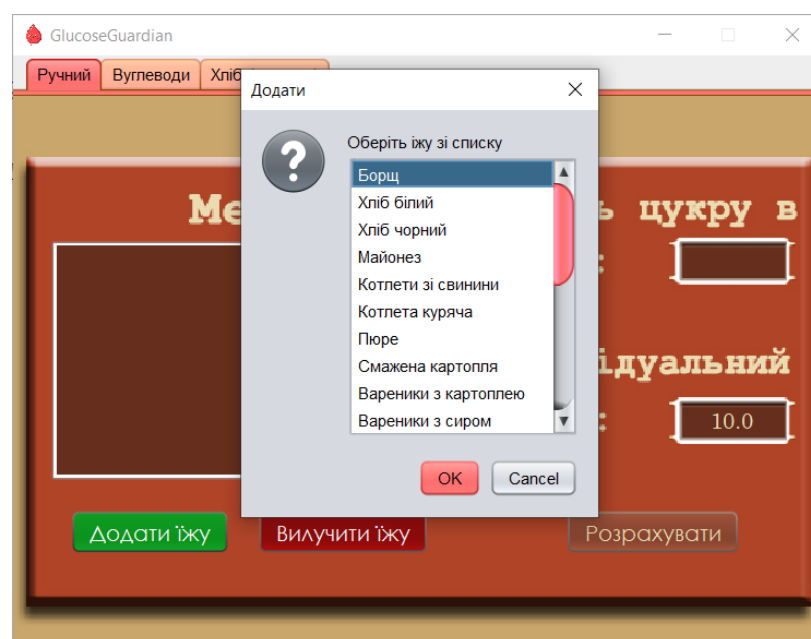


Рисунок 3.12 Дії програми при натисканні кнопки «Додати їжу»

Далі після того як користувач обирає страву з меню вона записується в змінну `n`, яка є об'єктом класу `String`. Потім виводиться ще один діалог вводу, в якому користувачу потрібно вказати кількість грам:

```
double m = Double.parseDouble(JOptionPane.showInputDialog("Кількість грам:"));
```

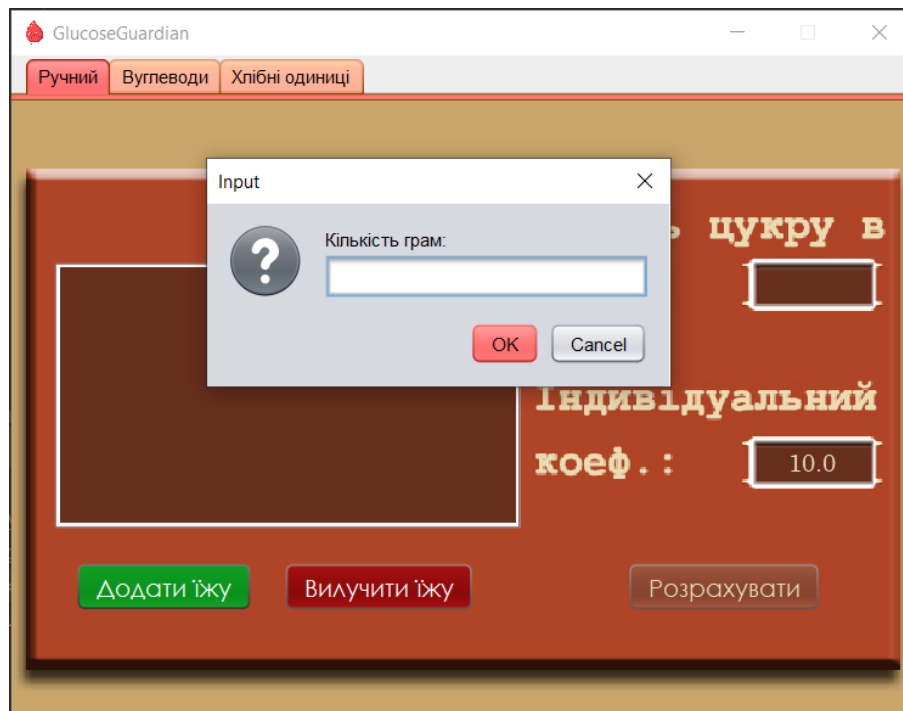


Рисунок 3.13 Програма просить користувача ввести кількість грам

Коли користувач ввів кількість грам, це число записується в змінну `m`, для подальших розрахунків.

```
sklad.add(n+" - "+m+" грам");
textArea.setText("");
sklad.forEach((e) ->{
    textArea.append(e+"\n");
});
```

Далі визивається метод класу `ArrayList` – `add`:

```
carbs.add(m*hashMenu(n));
```

В цей метод ми передаємо вираз який помножує кількість грам на кількість вуглеводів в одному грамі, їжі, яку обрав користувач. Для збереження цих

коефіцієнтів було обрано клас HashMap, в програмі він виступає заміною бази даних. В ньому зберігається ключ та число, ключем є назва страви, а числом коефіцієнт вуглеводів в їжі. Через метод hasMenu(n), програма звертається до цього методу та передає йому назву страви, яку користувач обрав раніше, та повертає число, яке відповідає ключу: `return menu.get(s)`.

Далі число грам, яке ввів користувач, помножується на коефіцієнт, та записується у динамічний масив `carbs`, який нам буде потрібен для розрахунків необхідної дози інсуліну.

Потім для виводу пункту меню, яке тільки-що користувач додав, було створено ArrayList з назвою `sklad`, та використаний метод `add`:

```
sklad.add(n+ " - "+m+" грам");
textArea.setText("");
sklad.forEach((e) ->{
    textArea.append(e+"\n");
});
```

Де `n` – назва страви, а `m` – кількість грам.

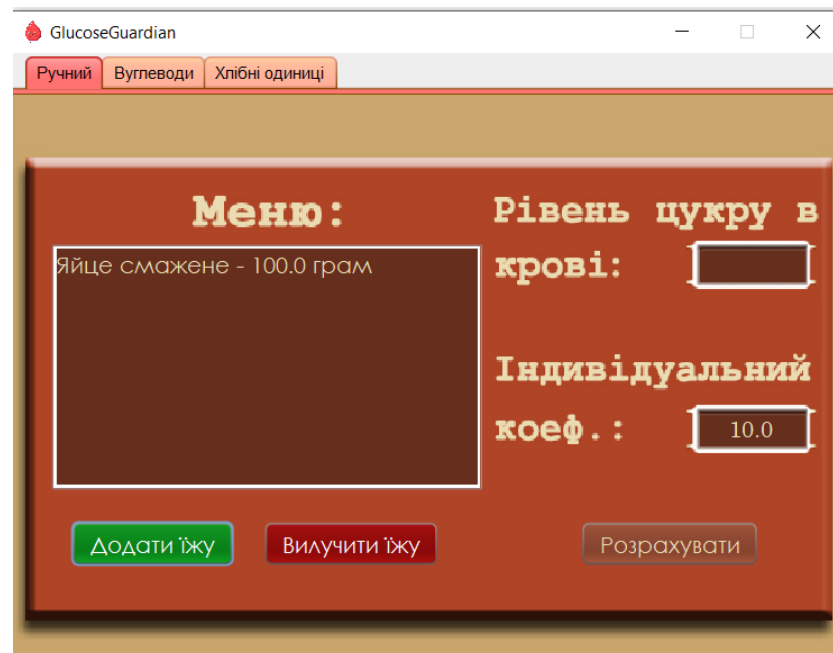


Рисунок 3.14 Приклад відображення пункту меню на екрані

Також через метод `setText`, `textArea` становиться пустим, після цього через метод `forEach` класу `ArrayList` відбувається відображення пунктів меню яке ввів користувач.

Весь цей метод, `addActionListener`, знаходиться у структурі `try-catch`. Це зроблено зад-ля перевірки на правильність введення даних від користувача:

```
catch (NumberFormatException e)
```

Якщо користувач в момент коли він вводить кількість грам продукту, значення буде не в числовому форматі, то користувач побачить повідомлення, яке зроблено через метод `JOptionPane.showMessageDialog()`:

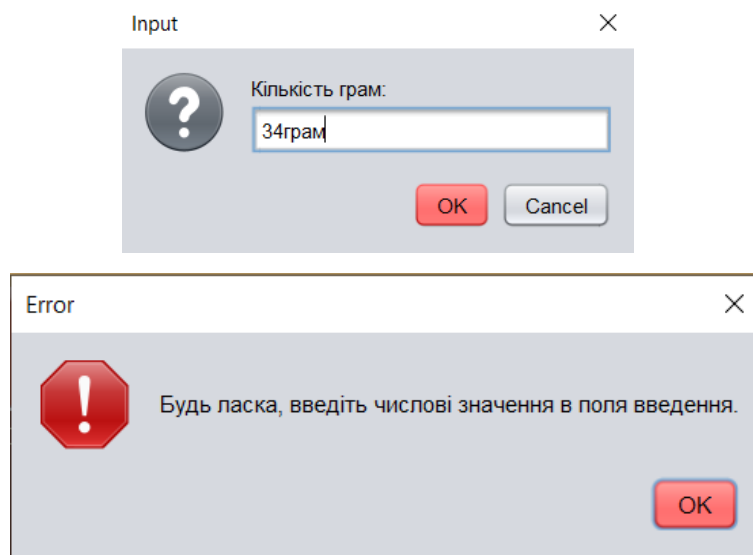


Рисунок 3.15 Помилка при введенні нечислового формату

Далі, якщо користувач зробив помилку, чи хоче видалити якийсь пункт в меню, він натискає на кнопку «Вилучити їжу». Визивається метод `addActionListener({...})` який виводить діалогове вікно:

```
String del = (String)JOptionPane.showInputDialog(null, "Оберіть їжу зі списку",  
"Видалити", JOptionPane.QUESTION_MESSAGE, null, temp, temp[0]);
```

Де `temp` – це створений тимчасовий масив елементів в який ми вносимо дані з `ArrayList sklad` з допомогою методу `toArray()`:

```
String[] temp = sklad.toArray(new String[0]);
```

Користувач обирає їжу, яку хоче видалити з меню. За допомогою випадаючого списку:

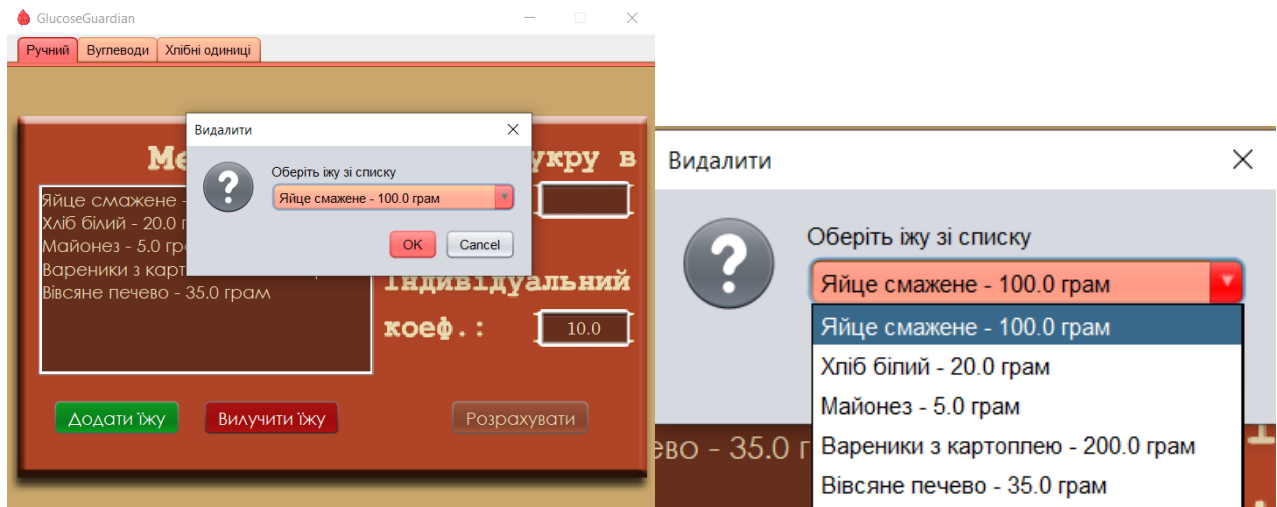


Рисунок 3.16 Вилучання їжі

Після того як користувач обрав що хоче видалити, програма записує цей пункт у змінну `del`, потім програма видаляє цей пункт з двох `ArrayList`'ів, які відповідають за кількість вуглеводів та за виведення меню на екран, це відбувається за допомогою методу `remove()` та `indexOf()` для видалення вуглеводів та потім наново виводить меню у `textArea`, вже без вилученого пункту:

```
carbs.remove(sklad.indexOf(del));
sklad.remove(del);
textArea.setText("");
sklad.forEach((e) ->{ textArea.append(e+"\n"); });
```

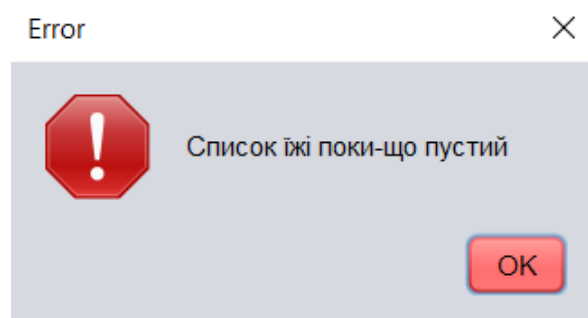


Рисунок 3.17 Помилка при пустому меню

Цей метод, `addActionListener`, зроблений через структуру `if-else`, де ми перевіряємо чи є в меню хоча б щось, якщо список пустий, користувачу виводить сповіщення(рисунок 3.17)

Далі, коли користувач заповнив меню, написав свій рівень цукру в крові та індивідуальний коефіцієнт, він натискає кнопку «Розрахувати», яка також зроблена через метод `addActionListener`.

При натисканні цієї кнопки, програма бере всю інформацію яка була надана користувачем та вираховує рекомендовану дозу інсуліну:

```
double sugar = Double.parseDouble(sugarField.getText());
double coeff = Double.parseDouble(coeffField.getText());
```

Так як текст введений у поле `textField` є об'єктом класу `String`, для розрахунків нам необхідно перетворити дані у змінну `double`, для цього використовується метод `parseDouble()`.

Також необхідно достати з масиву `carbs` суму елементів, щоб використати її у підрахунках:

```
double carb = 0.0;
for (int i = 0; i < carbs.size(); i++) {carb+= carbs.get(i);}
```

Далі для визначення необхідної дози інсуліну програма використовує формулу, яку було взято з відкритих джерел :

```
double dose = (carb/coeff)+((sugar* MMOL_TO_MGDL)-100)/40.0;
```

В цій формулі ми ділимо кількість вуглеводів на індивідуальний коефіцієнт чуттєвості до інсуліну (це необхідна доза для покриття спожитих вуглеводів) та додаємо до цього виразу необхідну дозу для зменшення цукру. В цій частині формули відбувається перетворення цукру у ммоль/л до мг/дл, 1 ммоль/л = 18.01559 мг/дл, це робиться для точності розрахунків. В формулі цифра 100 це рівень цукру до якого ця формула призведе, 100 мг/дл(5.5 ммоль/л) це норма цукру для людей з цукровим діабетом.

Після формули йде округлення до одного числа після коми:

```
dose = Math.round(dose * 10.0) / 10.0;
```

Це зроблено зад-ля зручності користувача.

Користувач побачить таке повідомлення(JOptionPane.showMessageDialog):

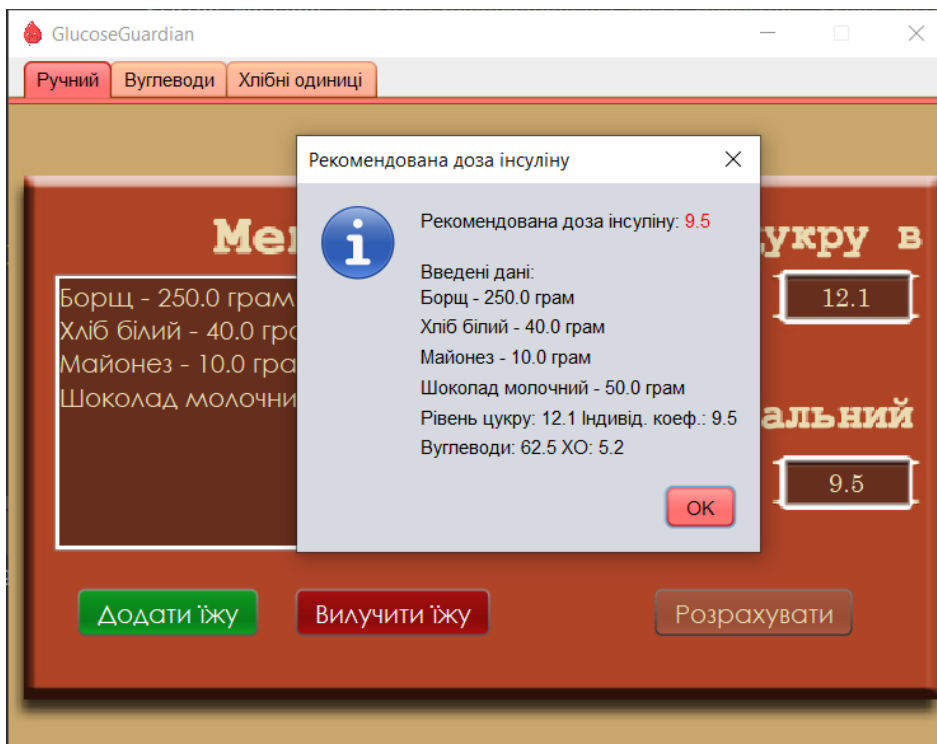


Рисунок 3.18 Повідомлення користувачеві з рекомендованою дозою інсуліну

Також метод цієї кнопки знаходиться у структурі try-catch (NumberFormatException). Завдяки цьому якщо користувач введе в поля рівня цукру в крові та/або індивідуальний коефіцієнт, йому буде показано сповіщення(рисунок 3.19).

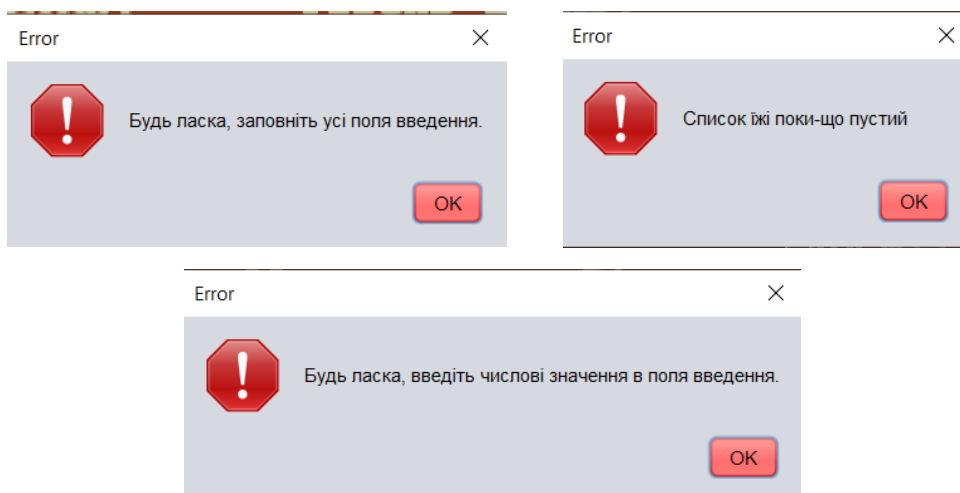


Рисунок 3.19 Помилки при введенні даних

Ще у методі кнопки, на самому початку, є перевірка на те, чи заповнене мені, та, якщо воно пусте, користувач побачить повідомлення(рисунок 3.19), також якщо поля рівень цукру та коефіцієнт не заповнені, також він побачить повідомлення про це(рисунок 3.19)

PersonalCarb та PersonalНо це два схожих класу, в яких відрізняє спосіб розрахунку, в першому використовуються вуглеводи, а в другому хлібні одиниці(ХО). Ці калькулятори будуть корисні користувачам які мають готові меню, в яких зазначена кількість вуглеводів/ХО.

Також в цих калькуляторів є ще одне корисне застосування – якщо зазначити кількість вуглеводів/ХО рівним 0, то програмне забезпечення покаже кількість інсуліну яка потрібна щоб рівень цукру в крові зменшився до рекомендованого (5,5 ммоль/л).

Вигляд цих класів був на рисунках 3.10 та 3.11. В кожному з них три поля вводу для рівня цукру в крові, індивідуального коефіцієнту та кількості вуглеводів/ХО та одне поле для рекомендованої дози інсуліну, в них показує результат прямо на інтерфейсі, немає виводу повідомлення як в класі Manual.

Також були додані кнопки «Info» та «Help» які допомагають користувачу зрозуміти як користуватись класами.

При натисканні на кнопку «Розрахувати» використовується такий же самий метод як в класі Manual, тільки тепер немає необхідності діставати з масиву вуглеводи, так як ми вже маємо необхідне число.

4. ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Таблиця 4.1 – Тест кейс ID: GG-01

Крок	Опис кроку	Тестові дані	Очікуваний результат
1	Додати в меню	«Борщ»	Програма просить ввести кількість грам
2	Ввід кількості грам	250	Пункт був доданий до меню
3	Додати в меню	«Майонез»	Програма просить ввести кількість грам
4	Ввід кількості грам	20	Пункт був доданий до меню
5	Додати в меню	«Хліб білий»	Програма просить ввести кількість грам
6	Ввід кількості грам	40	Пункт був доданий до меню
7	Вилучення з меню	«Борщ-250грам»	Пункт був вилучений з меню
8	Рівень цукру в крові	7.8	
9	Індивідуальний коеф.	9.2	
10	Кнопка «Розрахувати»		Програма виводить вікно в якому зазначені дані, які були введені та написана рекомендована доза: 3.2

Таблиця 4.2 – Тест кейс ID: GG-02

Крок	Опис кроку	Тестові дані	Результат
1	Додати в меню	«Борщ»	Програма просить ввести кількість грам
2	Ввід кількості грам	250	Пункт був доданий до меню
3	Додати в меню	«Майонез»	Програма просить ввести кількість грам

4	Ввід кількості грам	20г	Програма видає помилку в який вказано: «Будь-ласка введіть числові значення.» Пункт не був доданий до меню
5	Додати в меню	«Хліб білий»	Програма просить ввести кількість грам
6	Ввід кількості грам	40	Пункт був доданий до меню
7	Вилучення з меню	«Борщ-250грам»	Пункт був вилучений з меню
8	Рівень цукру в крові	12	
9	Індивідуальний коеф.	9.0	
10	Кнопка «Розрахувати»		Програма виводить вікно в якому зазначені дані, які були введені та написана рекомендована доза: 5.0

Таблиця 4.3 – Тест кейс ID: GG-03

Крок	Опис кроку	Тестові дані	Результат
1	Рівень цукру в крові	5.5	
2	Індивідуальний коеф.	10.0	
3	Кнопка «Розрахувати»		Програма виводить повідомлення що список їжі поки-що пустий

Таблиця 4.4 – Тест кейс ID: GG-04

Крок	Опис кроку	Тестові дані	Результат
1	Вилучення з меню		Програма виводить повідомлення що список їжі поки-що пустий

2	Додати в меню	«Яблуко»	Програма просить ввести кількість грам
3	Ввід кількості грам	100	Пункт був доданий до меню
4	Додати в меню	«Торт наполеон»	Програма просить ввести кількість грам
5	Ввід кількості грам	250	Пункт був доданий до меню
8	Рівень цукру в крові	9ммоль/л	
9	Індивідуальний коеф.	9.2	
10	Кнопка «Розрахувати»		Програма видає помилку в який вказано: «Будь-ласка введіть числові значення.»

Під час розробки програмного забезпечення було виконано тестування з використанням методів білого ящика, а також статичних перевірок коду та інших компонентів системи.

Під час тестування з використанням методу білого ящика було проаналізовано функції та компоненти програмного забезпечення з доступом до внутрішньої структури коду. Це дозволило виявити та виправити помилки, які можуть виникнути у роботі окремих функцій та їх взаємодії в системі в цілому. Також було проведено тестування різних шляхів виконання коду, з метою перевірки правильності його роботи та виявлення можливих вразливостей.

Під час статичних перевірок коду та інших елементів системи було виконано аналіз програмного коду, конфігураційних файлів та інших складових системи з метою виявлення можливих помилок та вразливостей на етапі розробки. Цей підхід дозволяє забезпечити виявлення помилок ще до того, як вони з'являться в процесі роботи системи у виробничому середовищі.

В результаті проведеного тестування було виявлено та виправлено ряд помилок, що забезпечить більш стабільну та безпечну роботу системи в майбутньому.

ВИСНОВКИ

Діабет є серйозним захворюванням, яке потребує постійного моніторингу рівня цукру в крові. Люди, які залежні від інсуліну, повинні регулярно вводити його в організм, щоб забезпечити належний рівень цукру. Однак, відслідковування цього рівня може бути складним завданням, особливо з урахуванням різних факторів, таких як харчування, фізична активність та стрес.

У цьому контексті, розробка програмного забезпечення для підтримки стабільного рівня цукру інсулінозалежних людей мовою Java є вкрай важливою. Розроблене програмне забезпечення допоможе інсулінозалежним людям контролювати свій рівень цукру в крові, що може поліпшити якість їхнього життя.

У процесі виконання дипломної роботи було проаналізовано різні типи діабету, його симптоми та можливі ускладнення. Були розглянуті різні методи введення інсуліну та вимірювання рівня цукру в крові. Крім того, було проведено дослідження та огляд різних програмних засобів, таких як Beetis, DiabTrend та mySugr. Цей аналіз дозволив визначити найбільш ефективні способи підрахунку рівня цукру в крові в залежності від спожитої їжі та вибрати оптимальні інструменти для розробки програмного забезпечення.

Для розробки програмного забезпечення, було обрано мову програмування Java, середовище програмування IntelliJ IDEA та бібліотеки Swing та AWT для розробки графічного інтерфейсу. Були сформульовані функціональні вимоги до програмного забезпечення, визначено алгоритм дій програми та створено графічний інтерфейс.

Після завершення розробки програмного забезпечення, було проведено тестування, під час якого було перевірено відповідність функціональних вимог та коректність роботи програми. Результати тестування були використані для виявлення та виправлення всіх помилок та недоліків програми.

Розроблене програмне забезпечення може бути цінним інструментом для людей з інсулінозалежним діабетом, яке допоможе їм ефективно контролювати рівень цукру в крові та покращити якість їхнього життя. Розробка цього

програмного забезпечення має велике значення для медичної практики і може суттєво покращити стан здоров'я та благополуччя людей, що страждають на це захворювання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. 14 ЛИСТОПАДА ВСЕСВІТНІЙ ДЕНЬ БОРОТЬБИ З ЦУКРОВИМ ДІАБЕТОМ [Електронний ресурс] // Центр Медичної статистики. – 2022. – Режим доступу до ресурсу: <https://zoiacms.zp.ua/2021/11/12/14-lystopada-vsesvitnii-den-borotby-z-tsukrovym-diabetom/>.
2. Лекція до всесвітнього дня діабету [Електронний ресурс] // Укрінформ. – 2021. – Режим доступу до ресурсу: <https://www.ukrinform.ua/rubric-society/3349150-na-cukrovij-diabet-hvorie-kozen-tridcatij-ukrainec-likar.html>.
3. Complications of diabetes [Електронний ресурс] // Емма Ноок. – 2019. – Режим доступу до ресурсу: <https://www.diabetes.org.uk/guide-to-diabetes/complications>.
4. Beetis [Електронний ресурс] – Режим доступу до ресурсу: <https://beetis.io/en>.
5. mySugr [Електронний ресурс] – Режим доступу до ресурсу: <https://www.mysugr.com/en/>.
6. DiabTrend [Електронний ресурс] – Режим доступу до ресурсу: <https://diabtrend.com/>.
7. Flash glucose monitors (freestyle libre) and continuous glucose monitors (CGM) [Електронний ресурс] // Diabetes UK. – 2022. – Режим доступу до ресурсу: <https://www.diabetes.org.uk/guide-to-diabetes/diabetes-technology/flash-glucose-monitors-and-continuous-glucose-monitors>.
8. Test strips and monitors [Електронний ресурс] // Diabetes UK. – 2019. – Режим доступу до ресурсу: <https://www.diabetes.org.uk/guide-to-diabetes/managing-your-diabetes/testing/test-strips-and-monitors>.
9. JayBourley G. differences between type 1 and type 2 diabetes [Електронний ресурс] / Gary JayBourley // Diabetes UK. – 2018. – Режим доступу до ресурсу: <https://www.diabetes.org.uk/diabetes-the-basics/differences-between-type-1-and-type-2-diabetes>
10. Diabetes treatments [Електронний ресурс] // Diabetes UK. – 2019. – Режим

доступу до ресурсу: <https://www.diabetes.org.uk/diabetes-the-basics/diabetes-treatments>.

11. IDF Diabetes Atlas 2022 Reports [Электронный ресурс] // International Diabetes Federation. – 2022. – Режим доступа до ресурсу: <https://diabetesatlas.org/2022-reports/>

12. IntelliJ IDEA features [Электронный ресурс] // JetBrains. – 2023. – Режим доступа до ресурсу: <https://www.jetbrains.com/idea/features/>.

13. Java Documentation [Электронный ресурс] // Oracle. – 2023. – Режим доступа до ресурсу: <https://docs.oracle.com/en/java/>.

14. NetBeans [Электронный ресурс] // Apache. – 2023. – Режим доступа до ресурсу: <https://blogs.apache.org/netbeans/>.

15. VS Code [Электронный ресурс] // Microsoft. – 2022. – Режим доступа до ресурсу: <https://code.visualstudio.com/docs/supporting/faq>.

ДОДАТОК А

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
 НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
 ТЕХНОЛОГІЙ
 КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ
 СТАБІЛЬНОГО РІВНЯ ЦУКРУ ІНСУЛІНОЗАЛЕЖНИХ ЛЮДЕЙ МОВОЮ JAVA

Виконав студент 4 курсу
 групи ПД-42
 Петренко Е.Д.
 Керівник роботи
 Трінтіна Наталія Альбертівна

Київ – 2023

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** - спрощення процесу визначення дози інсуліну в залежності від спожитої їжі, рівня цукру та індивідуального коефіцієнту.
- **Об'єкт дослідження** - процес підтримки стабільного рівня цукру інсулінозалежних людей
- **Предмет дослідження** - програмне забезпечення для підтримки стабільного рівня цукру

2

АНАЛІЗ АНАЛОГІВ



	Beetis	DiabTrend	mySugr
Платформи	Android	Android, iOS, Web	Android, iOS
Наявність української мови	-	-	-
Болюсний калькулятор в безкоштовній версії	-	-	+
Болюсний калькулятор в Платній версії	+	-	+
Доступне в Україні	+/-	+	-
Синхронізація з CGM	+	-	-
Можливість порахувати дозу інсуліну	-	-	-

3

ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1. Створити можливість швидко та зручно розрахувати дозу інсуліну
2. Можливість користувача розраховувати рекомендовану дозу інсуліну з різних вхідних даних
3. Включення в формулу дози індивідуального коефіцієнту
4. Розрахунок необхідної дози щоб зменшити рівень цукру та покрити споживані вуглеводи
5. Реалізація меню в додатку, щоб користувач обирав їжу, яку хоче спожити, а програма сама рахувала скільки в цій їжі вуглеводів, та давала необхідну дозу інсуліну.

4

ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



Java - це об'єктно-орієнтована мова програмування, яка використовується для розробки різноманітних програм та додатків на різних платформах.



Swing - це набір інструментів для створення графічного інтерфейсу користувача (GUI) у Java, що дозволяє створювати вікна, кнопки, меню та інші елементи GUI.



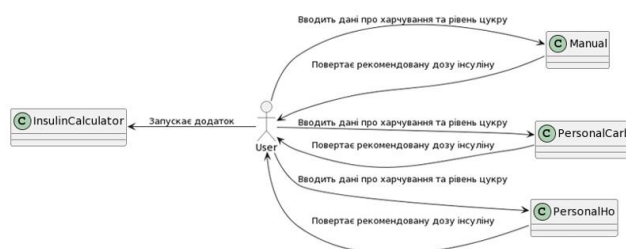
IntelliJ IDEA - це інтегроване середовище розробки (IDE), призначене для написання програм на Java, що підтримує велику кількість функцій та забезпечує продуктивність розробки.



AWT (Abstract Window Toolkit) - це перевірений часом інструментарій для створення графічного інтерфейсу користувача (GUI) у Java, що включає в себе набір компонентів GUI для розробки додатків.

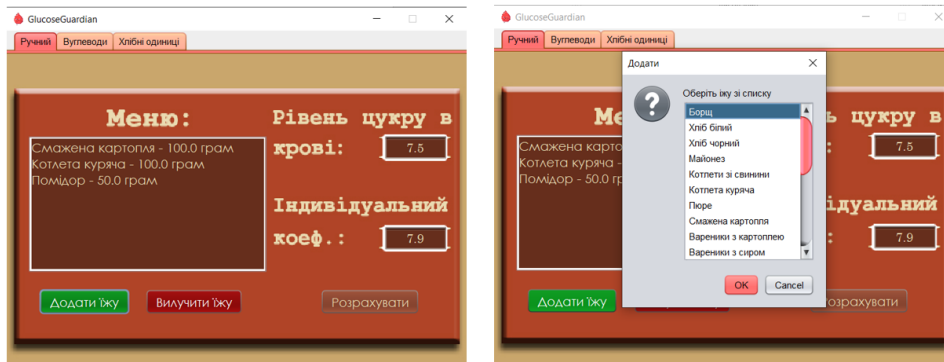
5

Схема взаємодії користувача з системою



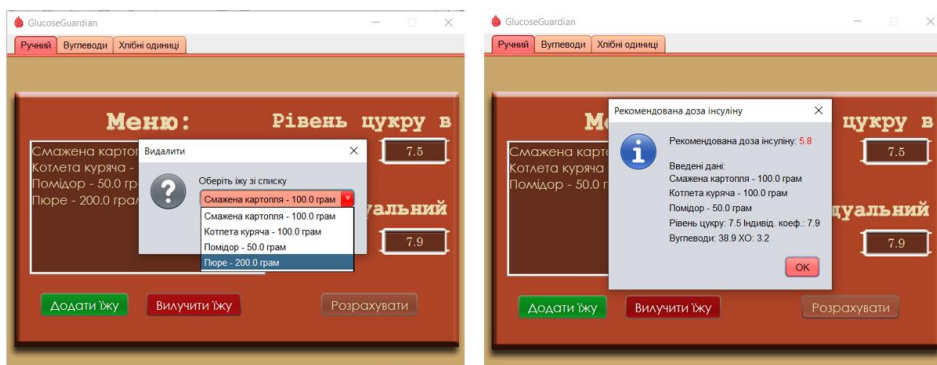
6

ЕКРАННІ ФОРМИ



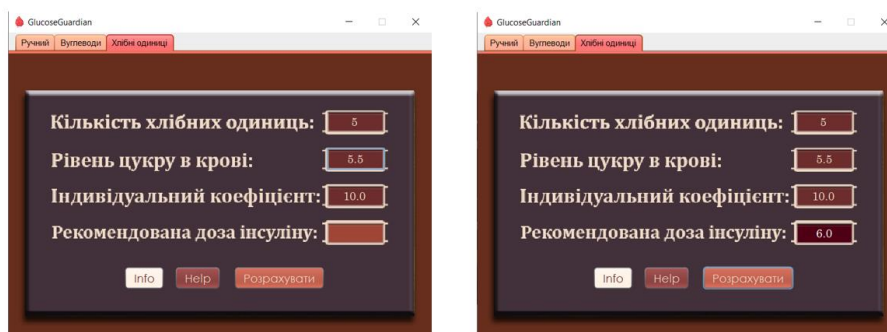
10

ЕКРАННІ ФОРМИ



11

ЕКРАННІ ФОРМИ



12

АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

1. Петренко Е.Д. Використання ІКТ для покращення діагностики та лікування діабету/ Н.А. Трінтіна, Е.Д Петренко// Всеукраїнська науково-технічна конференція «Застосування програмного забезпечення в ІКТ». Збірник тез. 20.04.2023, ДУТ, м. Київ — К.: ДУТ, 2023. — С. 69.
2. Петренко Е.Д. Роль ІоТ у покращенні лікування діабету/ Н.А. Трінтіна, Е.Д Петренко// IV Науково-технічна конференція «Сучасний стан та перспективи розвитку ІоТ». Збірник тез. 07.04.2023, ДУТ, м. Київ — К.: ДУТ, 2023. — С. ?.

13

ВИСНОВКИ

1. Проведено аналіз предметної галузі, за результатами якої було виявлено що нестабільний рівень цукру є дуже небезпечною проблемою, та сприяє загостренню хвороби, тому було прийняте рішення по створенню ПЗ для розрахунку необхідної дози інсуліну в залежності від спожитої їжі.
2. Проведено дослідження засобів створення програмного забезпечення.
3. Зроблено концепцію ПЗ та вимоги, UML-діаграми класів, станів, діяльності.
4. Розроблено програмне забезпечення яке працює на JRE, та може бути запущено всюди де встановлено JVM.
5. Зроблений простий та інтуїтивно зрозумілий інтерфейс, для швидкого використання користувачем
6. Розроблена можливість рахувати рекомендовану дозу інсуліну в залежності від спожитої їжі, рівня цукру та індивідуального коефіцієнту.

14

ДЯКУЮ ЗА УВАГУ!

