

**CENTRUL DE EXCELENȚĂ ÎN  
INFORMATICĂ ȘI TEHNOLOGII  
INFORMATIONALE**

**СПЕЦИАЛЬНОСТЬ «Planificarea aplicatiilor Desktop »**

**Индивидуальная работа 1(4 вр.)**

**Тема: Приложение для заметок**

**Преподаватель: Gairunova Natalia**

**Учащийся: Matusar Dmitrii**

**Кишинев, 2025**

# Содержание

1. Описание работы
2. Модули приложения
3. SRS — функциональные и нефункциональные требования
4. Диаграммы UML
5. Прототип Figma
6. Репозиторий Git/GitHub
7. Заключение
8. Использованные источники

## **1. Описание работы**

Данное приложение «Notes» представляет собой настольное приложение для создания, организации и просмотра заметок. Пользователь может создавать новые заметки, выбирать категорию, просматривать список всех заметок, а также использовать тёмную и светлую тему интерфейса. Основная цель приложения — обеспечить удобное хранение информации.

Целевые пользователи: студенты, офисные сотрудники, пользователи, которым необходимо быстро сохранять и структурировать информацию.

## **2. Модули приложения**

- Модуль просмотра всех заметок
- Модуль категорий (Work, Study, Projects, Assignments, Wishlist, Videos)
- Модуль создания новой заметки
- Модуль чтения заметки
- Модуль выбора темы оформления (тёмная/светлая)
- Модуль быстрого поиска

## **3. SRS — функциональные и нефункциональные требования**

### **Функциональные требования:**

- Пользователь может создавать заметку.
- Пользователь может просматривать заметки.
- Можно выбирать категорию заметки.
- Реализован быстрый поиск.
- Можно поделиться заметкой.

### **Нефункциональные требования:**

- Данные пользователя должны сохраняться автоматически при наличии соединения с интернетом.
- Интерфейс должен быть удобным и минималистичным.

- Реализация тёмной и светлой темы.
- Быстрая работа приложения и моментальный отклик интерфейса.

## 4. Диаграммы UML

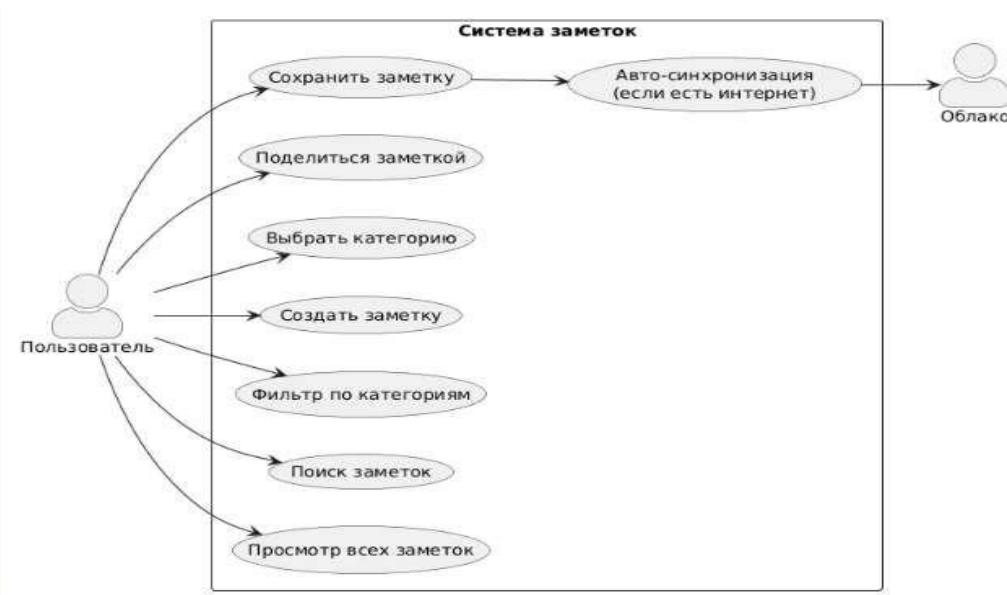
В процессе разработки приложения «Notes» были созданы две UML-диаграммы, отражающие основные процессы работы приложения: диаграмма вариантов использования (Use Case) и диаграмма деятельности (Activity). Эти диаграммы позволяют визуализировать взаимодействие пользователя с системой и последовательность операций внутри приложения.

### Диаграмма Use Case

Диаграмма Use Case отражает функциональность приложения с точки зрения пользователя.

В системе присутствует один основной актор — **Пользователь**, который может выполнять следующие действия:

- Просматривать все заметки
- Выбирать категорию заметок
- Искать заметку через строку поиска
- Создавать новую заметку
- Читать содержимое заметки
- Делиться заметкой через меню «Share»



**Код диаграммы:**

```
@startuml
```

```
actor Пользователь
```

```
Пользователь --> (Просмотр всех заметок)
```

```
Пользователь --> (Выбор категории)
```

```
Пользователь --> (Поиск заметки)
```

```
Пользователь --> (Создать новую заметку)
```

```
Пользователь --> (Открыть заметку)
```

```
Пользователь --> (Поделиться заметкой)
```

```
@enduml
```

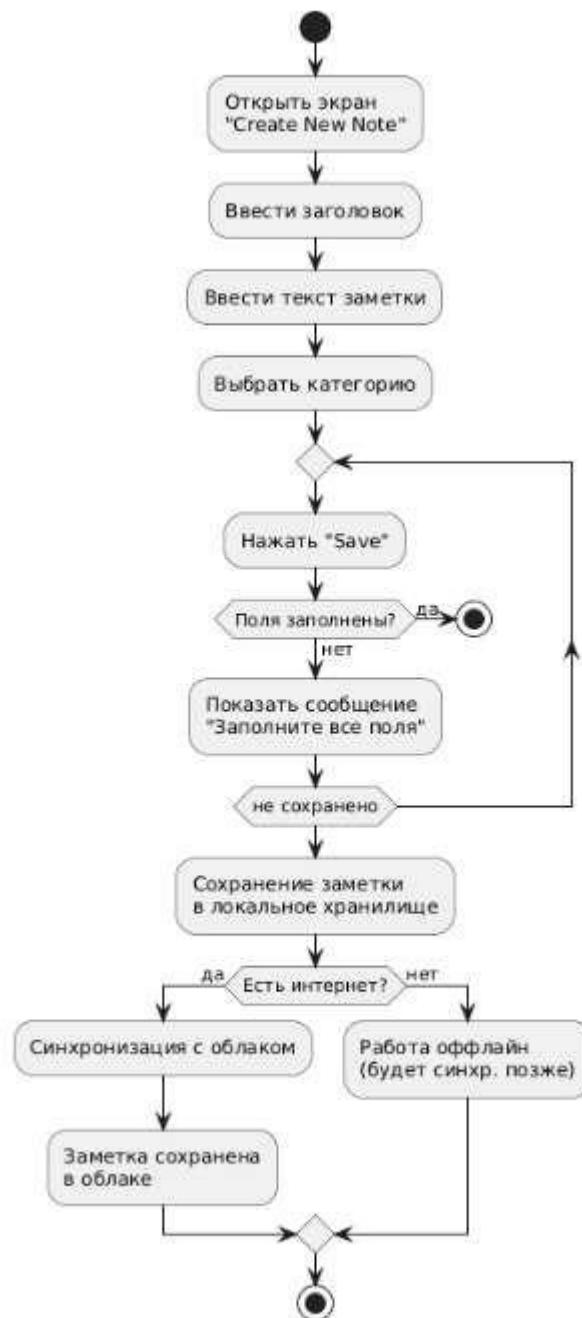
**Диаграмма Activity:**

Диаграмма Activity описывает процесс сохранения заметки в приложении.

Поскольку приложение автоматически синхронизирует данные с облаком (при наличии соединения с интернетом), этот процесс показан в виде последовательности действий.

Алгоритм:

1. Пользователь создаёт заметку
2. Система проверяет наличие интернета
3. Если интернет есть → заметка сохраняется локально и отправляется в облако
4. Если интернета нет → заметка сохраняется локально
5. Синхронизация произойдёт автоматически при следующем подключении



**Код диаграммы:**

```
@startuml  
start  
    :Создание новой заметки;  
    :Проверка наличия интернета;  
  
    if (Интернет есть?) then (Да)  
        :Сохранить заметку локально;  
        :Отправить данные в облако;  
    else (Нет)  
        :Сохранить заметку локально;  
    endif  
  
stop  
@enduml
```

## 5. Прототип Figma

<https://www.figma.com/design/B68smatjRYbU2KKeeKJSKH/Untitled?m=auto&t=SUxy0CwUN2x7KnPd-1>

## 6. Репозиторий Git/GitHub

Создан репозиторий приложения. Использованные команды Git:

```
git init  
git add .  
git commit -m "Initial commit"  
git branch -M main  
git remote add origin <ссылка_на_репозиторий>  
git push origin main
```

<https://github.com/MatusarDima/notes-ceiti>

## 7. Заключение

В ходе выполнения данной работы был разработан дизайн настольного приложения «Notes» с поддержкой категорий, удобным интерфейсом, светлой и тёмной темами. Реализован концепт работы приложения, представлены диаграммы UML и создан репозиторий GitHub.

## 8. Использованные источники

- figma.com
- plantuml.com
- github.com
- Материалы курса СЕИТ