

ТЕХНОЛОГИЧНО УЧИЛИЩЕ ЕЛЕКТРОННИ СИСТЕМИ
към ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ

ДИПЛОМНА РАБОТА

Тема: Библиотека за създаване на текстов потребителски интерфейс

Дипломант:
Ивайло Генчев

Научен ръководител:
маг. инж. Владимир Алексиев

С О Ф И Я

2 0 2 2

Увод

Първоначално, когато компютрите започват масово да се разпространяват, те могат да се достъпват единствено през терминал - текстови интерфейс, с който се работи с клавиатура. Не след дълго, поради бързия технологичен напредък, графичния потребителски интерфейс (ГПИ) бива създаден заедно с първата компютърна мишка. Това развитие значително увеличава достъпността и лекотата на използване на компютърните устройства.

Днес, около половин век по-късно, почти всеки притежава устройство с ГПИ. Текстовият потребителски интерфейс (ТПИ) се използва доста по-рядко - предимно в среди, където няма ГПИ, например при сървъри с ограничени ресурси или от потребители с големи възможности. Поради тази причина повечето съществуващи програми за ТПИ съществуват от десетилетия, а новите са принудени да използват софтуер, разработен преди десетки години. Това не намалява функционалността на програмите, но увеличава времето им за разработка и може да ги направи по-недостъпни за потребителя. Този проект решава именно този проблем.

За цел на проекта е поставена разработката на лесна за ползване библиотека, която да позволява създаването на модерен потребителски интерфейс.

Задачи

- Логическо отделени елементи на потребителския интерфейс.
- Оформление на компонентите на потребителския интерфейс.
- Обработка на събития от мишка и клавиатура.

В **Глава 1** е направен обзор на средите на работа на ТПИ и методите за управлението им. Също така е направен преглед на съществуващи библиотеки, които се използват от приложения и разработчици по целия свят.

Глава 1

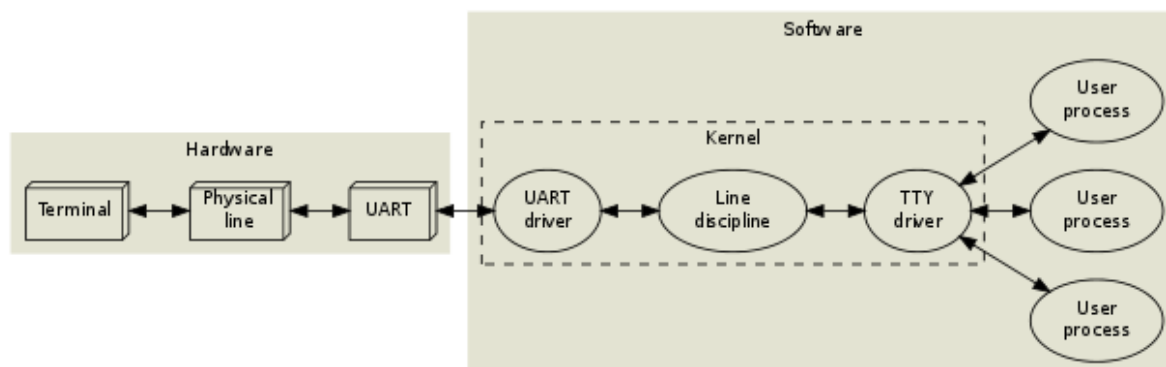
Проучване

1.1 Терминали

От библиотеката се изисква създаването на текстови потребителски интерфейси, което зависи от средата, в която се изпълнява програмата. Разгледани са възможните среди без съсредоточаване в детайлите.

1.1.1 Телетайп терминал - TTY

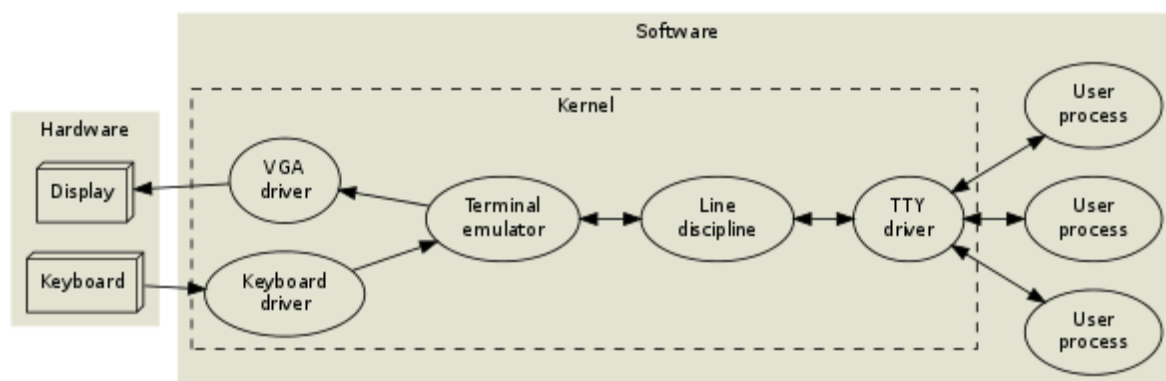
Този вид терминал се нарича така, поради сходността си с телеграфния апарат. На фигура 1.1 е показана блокова схема на принцип на работа на TTY терминал. Потребителят пише на хардуерния терминал, който е свързан към UART (универсален асинхронен приемник и предавател) на компютъра. Операционната система на компютъра съдържа UART драйвер, който управлява физическото предаване на байтове. "Дисциплината на линията" определя командите за редактиране на терминала - например изтрий линия, препечатай линия, изтрий дума.



Фигура 1.1: Принцип на работа на хардуерен TTY

1.1.2 Псевдотелетайп терминал - PTY

Фигура 1.3 постига същия резултат като фигура 1.1, като разликата е, че хардуерен терминал вече не се използва. Linux ядрото използва софтуерно реализиран краен автомат, който емулира същите функционалности.



Фигура 1.2: Принцип на работа на PTY

1.1.3 Терминален емулатор

Терминалните емулатори използват PTY подсистемата, но за разлика от PTY работят на ring 3 (потребителско пространство). Това ниво на абстракция добавя много повече гъвкавост, но и усложнява имплементацията си, защото един PTY може да се отвори в друг PTY.

Тази гъвкавост води до разлики в терминалните емулатори. Част от тези, които са програмирани преди въвеждането на ISO 6429, не го имплементират, а други имплементират разширени CSI кодове

1.2 Контролни символи

Контролните символи, също наричани непечатни, нямат графично представяне, а се използват за управление на устройства и предаване на данни. POSIX стандартът дефинира само осем контролни символа, от които по-често използваните са:

- `\0` - край на символен низ
- `\n` - нов ред
- `\t` - табулация
- `\r` - връщане на каретата

1.2.1 ANSI контролна последователност

ANSI кодовете могат да въздействат на поведението на терминала, като променят позицията на курсора, цвета на фона и символите, стилизирането на шрифта или други настройки.

Структурата на всички ANSI кодове е следната:

CSI [байтове на режима] n1;n2... [междинни байтове] финален байт

ANSI последователността винаги започва с CSI и приключва с финален байт. Байтовете на режима могат да бъдат в обхват от '0' (0x30) до '?' (0x3f), а междинните байтове в обхват от ' ' (0x20) до '/' (0x2f). Числата n1, n2, ... не са задължителни и, ако стойността им не е посочена, се приемат за 0 или 1 в зависимост от операцията, която ще се изпълни заради ANSI кода.

Въпреки че е позволено наличието на повече от един междинен байт и байт на режима, това не се ползва.

Идентификатор на контролна последователност (CSI)

Преамбюл, с който се разпознава началото на ANSI кода. Има големина 2 байта, като започва с ESC символ (0x1b) и "["(0x5b)

1.3 Преглед на съществуващи TUI библиотеки

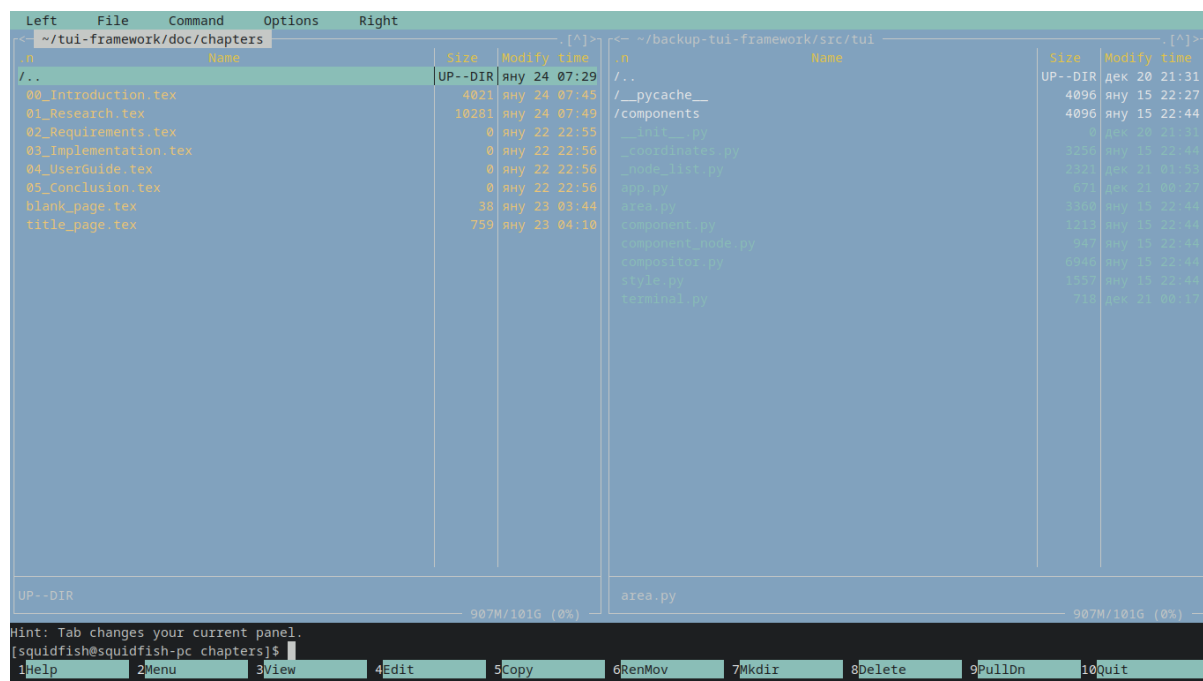
1.3.1 ncurses

ncurses е най-използваната библиотека за управление на терминали в UNIX-подобни среди, позволяваща създаване на приложения с текстов потребителски интерфейс.

С ncurses програмистите не трябва да се грижат за използването на правилните контролни символи при различните терминали. Библиотеката предоставя и абстракция за логическо отделение на компоненти - прозорци. Прозорците се съпоставят с координатите на терминала, като всеки прозорец представлява масив от символи, в който програмистът

може да промени външния вид на прозореца.

Една от най-силните страни на ncurses е съвместимостта на библиотеката. Тя може да работи във всякакви среди при всякакви условия



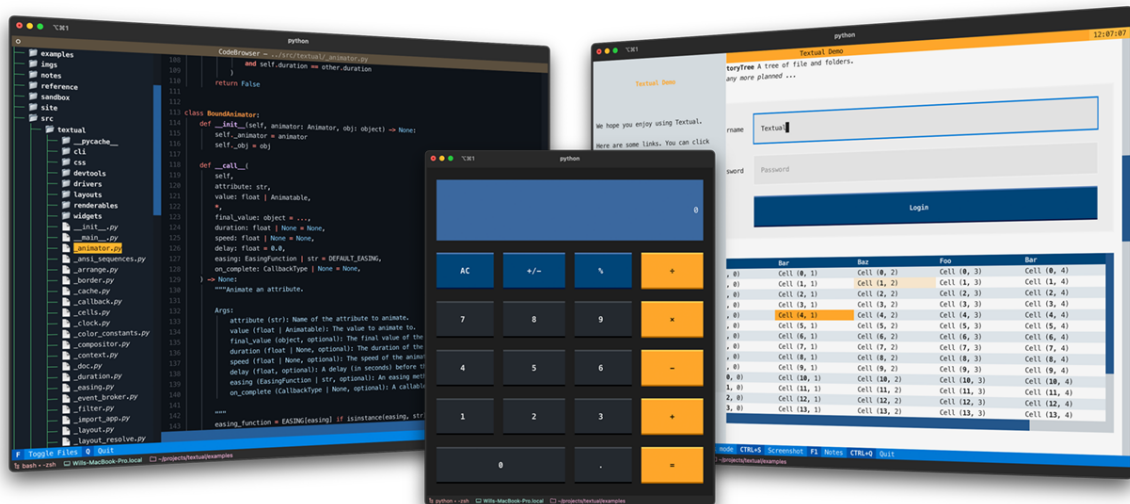
Фигура 1.3: midnight commander - файлов мениджър написани с ncurses

1.3.2 Textual

Понастоящем Textual е авангардна библиотека за създаване на модерни текстови потребителски интерфейси.

Textual взема вдъхновение от уеб разработката. В Библиотеката нивото на абстракция е много по-високо отколкото при ncurses. Използват се компоненти, които са подредени в дървовидна структура и силно наподобяват HTML елементи. Компонентите могат да се стилизират със CSS.

Има поддръжка на мишка и могат да се създават плавни анимации с над 60 кадъра в секунда (в зависимост от терминалния емулатор).



Фигура 1.4: Примерни приложения написани с Textual

Глава 2

Глава 2

Глава 3

Глава 3

Глава 4

Глава 4

Глава 5

Заключение

Съдържание

1	Проучване	5
1.1	Терминали	5
1.1.1	Телетайп терминал - TTY	5
1.1.2	Псевдотелетайп терминал - PTY	6
1.1.3	Терминален емулятор	7
1.2	Контролни символи	7
1.2.1	ANSI контролна последователност	7
1.3	Преглед на съществуващи TUI библиотеки	8
1.3.1	ncurses	8
1.3.2	Textual	9
2	Глава 2	10
3	Глава 3	11
4	Глава 4	12
5	Заключение	13