

# Pawculator

Femboys Polska 

Dominik Huml  
Matyáš Krejza  
Tomáš Brablec  
Matouš Jašek

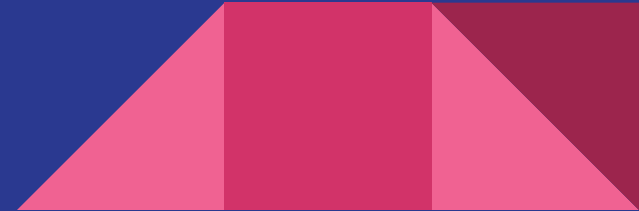
## 0.5 - Agenda:



Tech stack  
Features

Front end  
Backend  
Graphics  
User manual

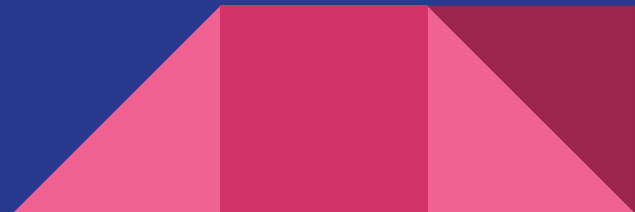
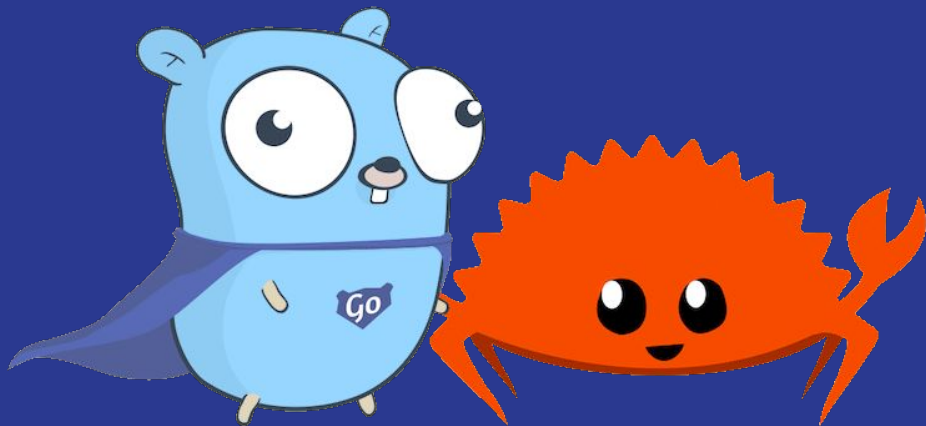
Future features  
Problem Evaluation



## 0.6 - Disclaimer



Při tvorbě projektu nebylo ublíženo žádnému zvířátku. Ani modrým křečkům, ani humrům, krabům či jiným vodním korýšům.

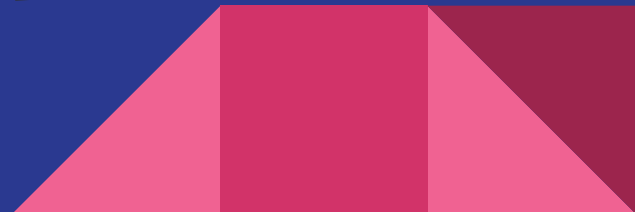
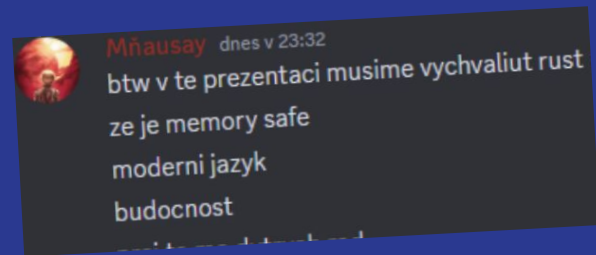
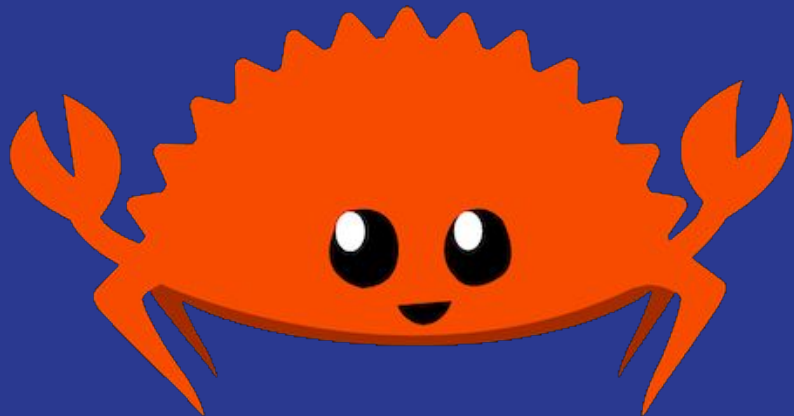


# 1 - Tech stack - Language



Rust – Modern, multi pragmatic, memory safe language

- C like performance
- Revolution in memory management

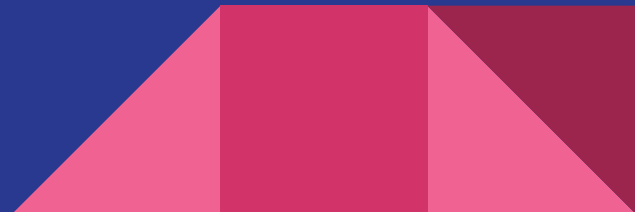
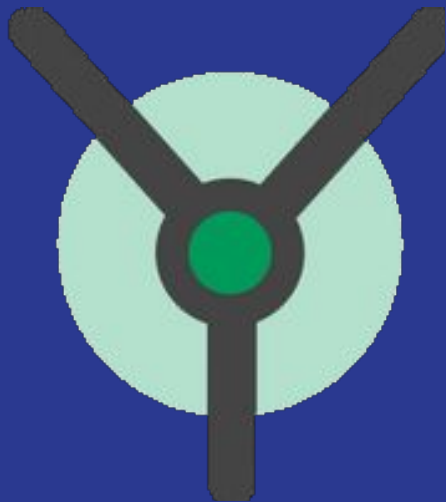




# 1.1 - Tech stack - FE

## Yew

- Component Based web framework
- Hot reloading, SSR, JavaScript interoperability

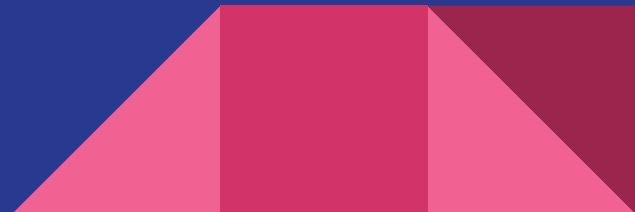
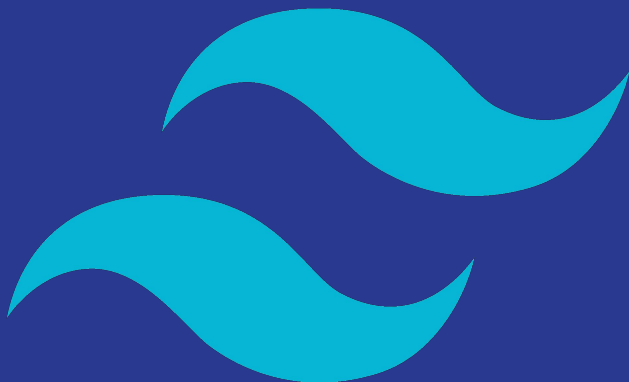




## 1.2 - Tech stack - FE

### Tailwind

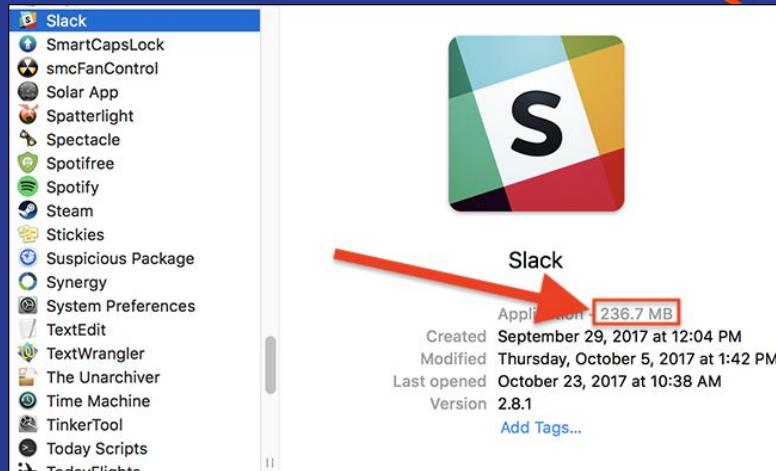
- A utility-first CSS framework
- Easy design system, unified css code



## 1.3 - Builder

### Tauri

- Cross-platform build tool
- Easy to use program bundling
- Electron like (but better)

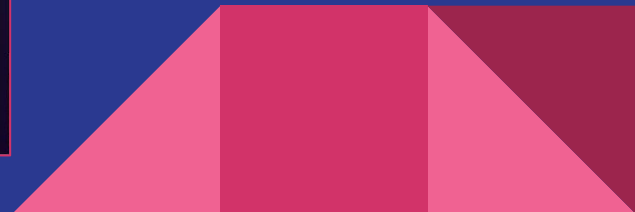
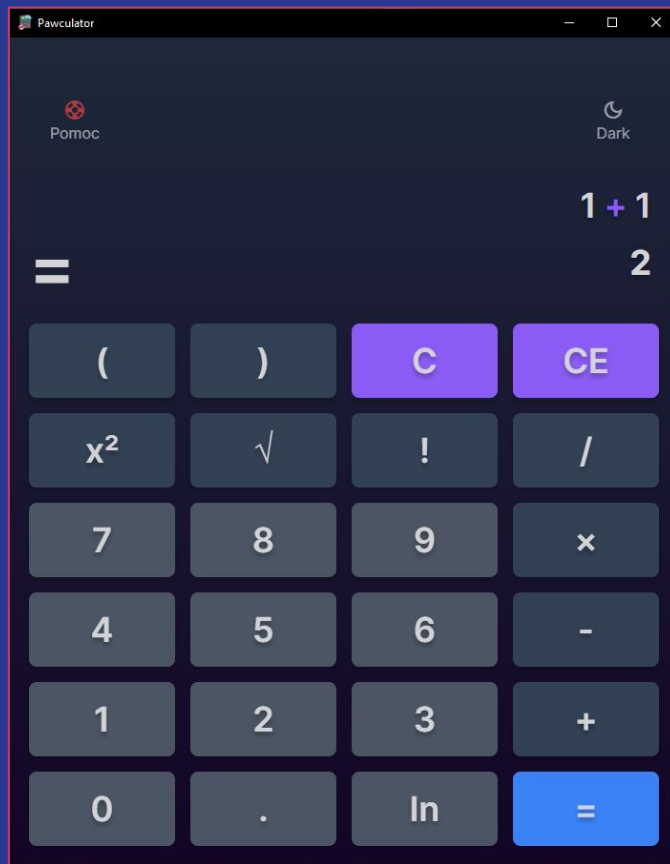


```
kiklop in 🌐 kentus in ivs2/xbrabl04_xhumld00_xjasek18_xkrejz07/install  
> ls
```

| # | name                              | type | size     | modified    |
|---|-----------------------------------|------|----------|-------------|
| 0 | <u>calculator_0.1.0_amd64.deb</u> | file | 4.7 MB   | 3 weeks ago |
| 1 | <u>stddev_0.1.0_amd64.deb</u>     | file | 151.3 KB | 3 weeks ago |

# 1.4 - Features

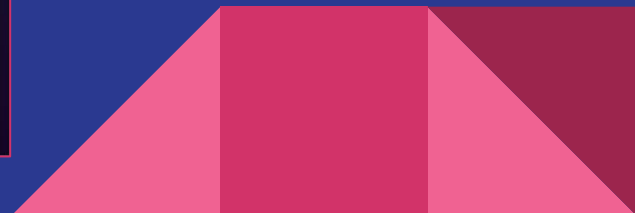
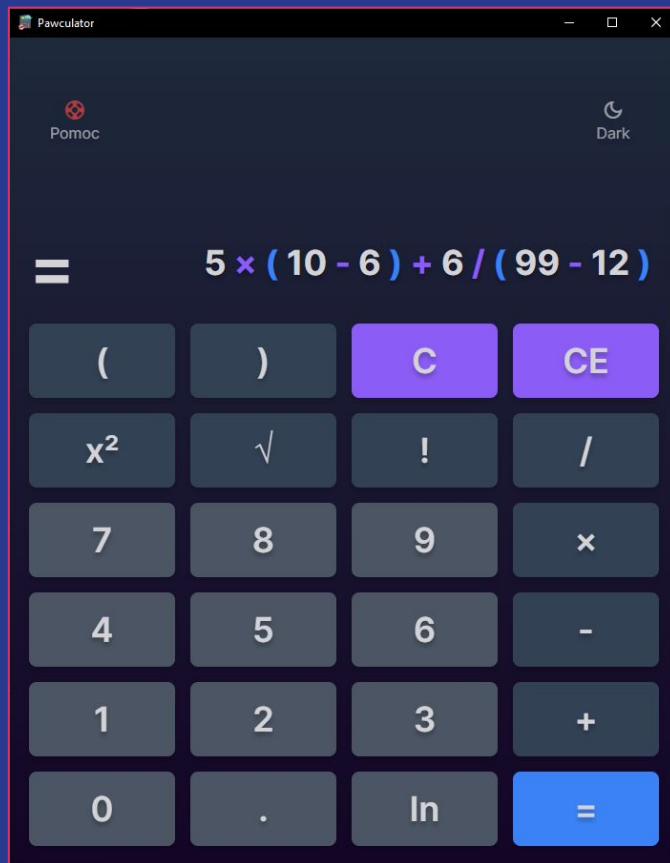
Calculator 👍





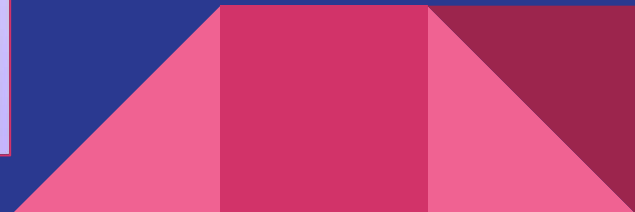
# 1.4 - Features

Expression parser



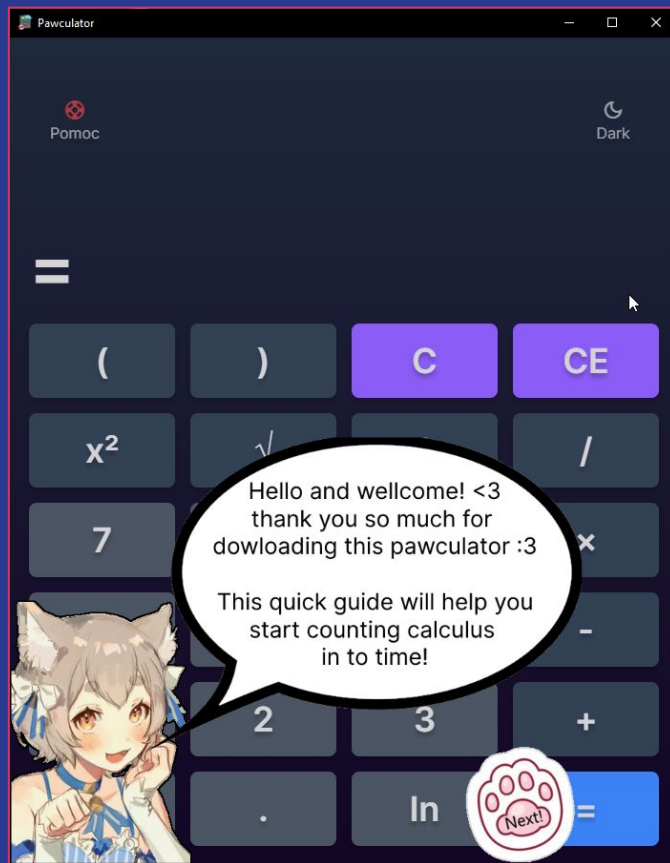
# 1.4 - Features

Dark/light mode 🌙



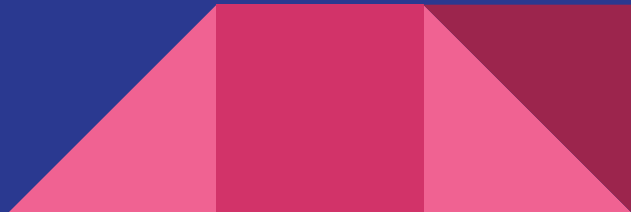
# 1.4 - Features

Catboy helper 🐱





Mark Russinovich - Microsoft Azure CTO





## 2.1 - Backend

- Automatic expression parser
- Math library

```
impl MathExpr {  
    /// Recursively evaluates given expression and returns `Some` of value  
    /// or `None`, if expression is invalid (such as division by zero, log of negative  
    /// value, arithmetic overflow, etc).  
    pub fn eval(&self) → Option<Decimal> {  
        match self {  
            MathExpr::AddExpr(a, b) => {  
                let (a, b) = (a.eval(), b.eval());  
                if let (Some(a), Some(b)) = (a, b) {  
                    a.checked_add(b)  
                } else {  
                    None  
                }  
            }  
  
            MathExpr::SubExpr(a, b) => {  
                let (a, b) = (a.eval(), b.eval());  
                if let (Some(a), Some(b)) = (a, b) {  
                    a.checked_sub(b)  
                } else {  
                    None  
                }  
            }  
        }  
    }  
}
```



## 2.1 - Backend

```
thing = _{number | "(" ~ subexpr ~ ")" }

prefix_op = _{
  minus_sign
  | ln_op
  | abs_op
  | sqrt_op
}
minus_sign = { "-" }
ln_op = { "ln" }
abs_op = { "abs" }
sqrt_op = { "√" }

postfix_op = _{ factorial_op }
factorial_op = { "!" }

unary_subexpr = _{ prefix_op* ~ thing ~ postfix_op* }
```



## 2.2 - Frontend

- Component based approach
- State management

```
html! {  
    
  <div class={classes!("app", "bg-gradient-to-b", color_theme, "h-screen", "p-5", "flex", "flex-col")}>  
    <div class={classes!("flex-none", "flex", "justify-between", "items-start", "py-5")}>  
      <HelpIcon/>  
      <ThemeSwitcher/>  
    </div>  
    <Display/>  
    <Keypad/>  
    if state.show_helper {  
      <HelperAvatar/>  
    }  
  </div>  
}
```

## 2.2 - Frontend



```
/// Add array of values value to an expression
pub fn expression_add_many(state: &mut AppState, values: Vec<&str>) {
    for value in values.into_iter() {
        expression_add(state, value);
    }
}

/// Add value to an expression
pub fn expression_add(state: &mut AppState, value: &str) {
    let last_term = state.expression.last();

    if last_term.is_none() {
        state.expression.push(value.to_owned());
        return;
    }

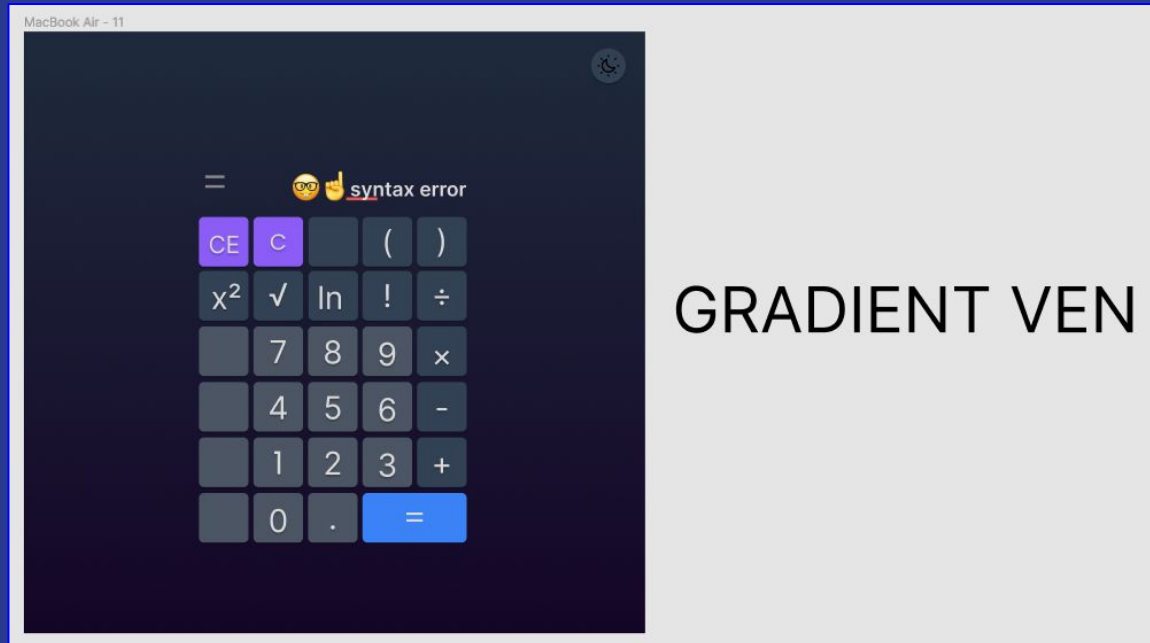
    let last_term = last_term.unwrap();

    if is_number(&value.to_owned()) && is_number(last_term) {
        if let Some(mut last_item) = state.expression.pop() {
            // If last term is a number, and current term is number too, we concatenate them
            last_item.push_str(&value);
            state.expression.push(last_item);
        }
    } else {
        state.expression.push(value.to_owned());
    }
}
```

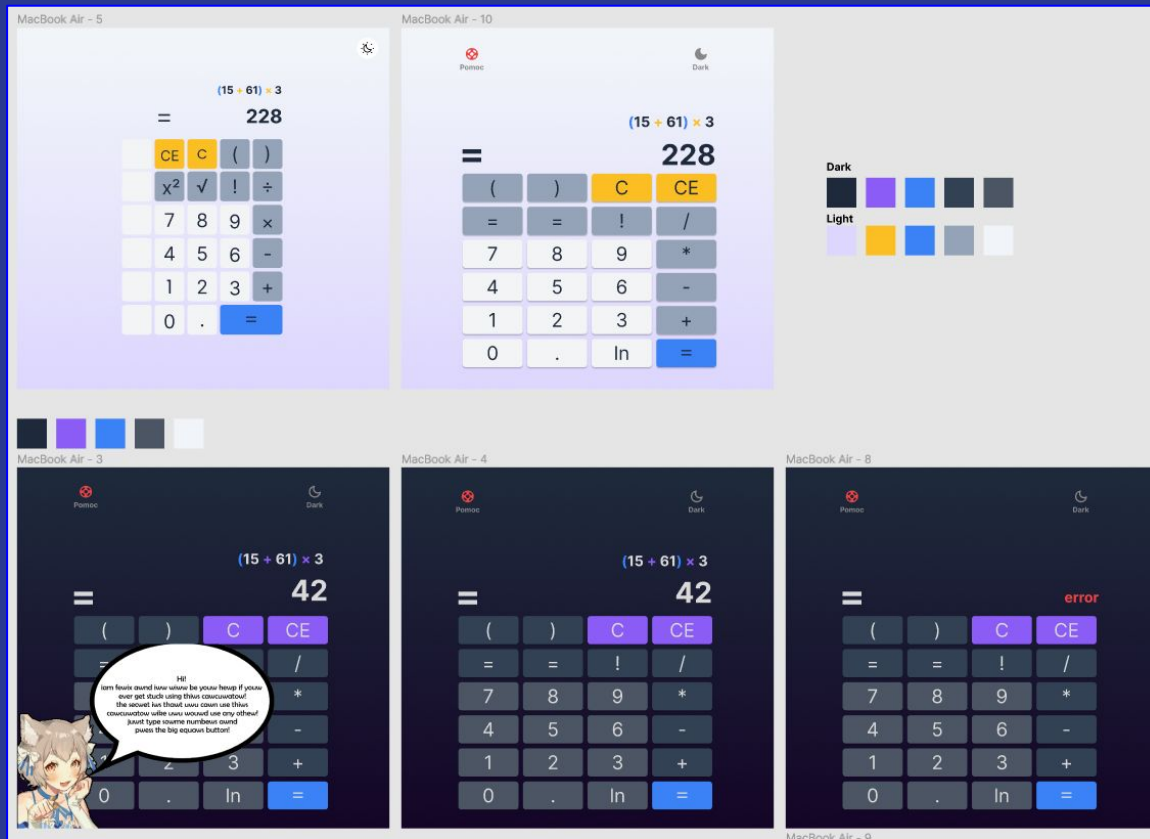


## 2.3 - Graphics

- Multiple designs



## 2.3 - Graphics



## 2.4 - User guide



### Uživatelská příručka ke kalkulačce

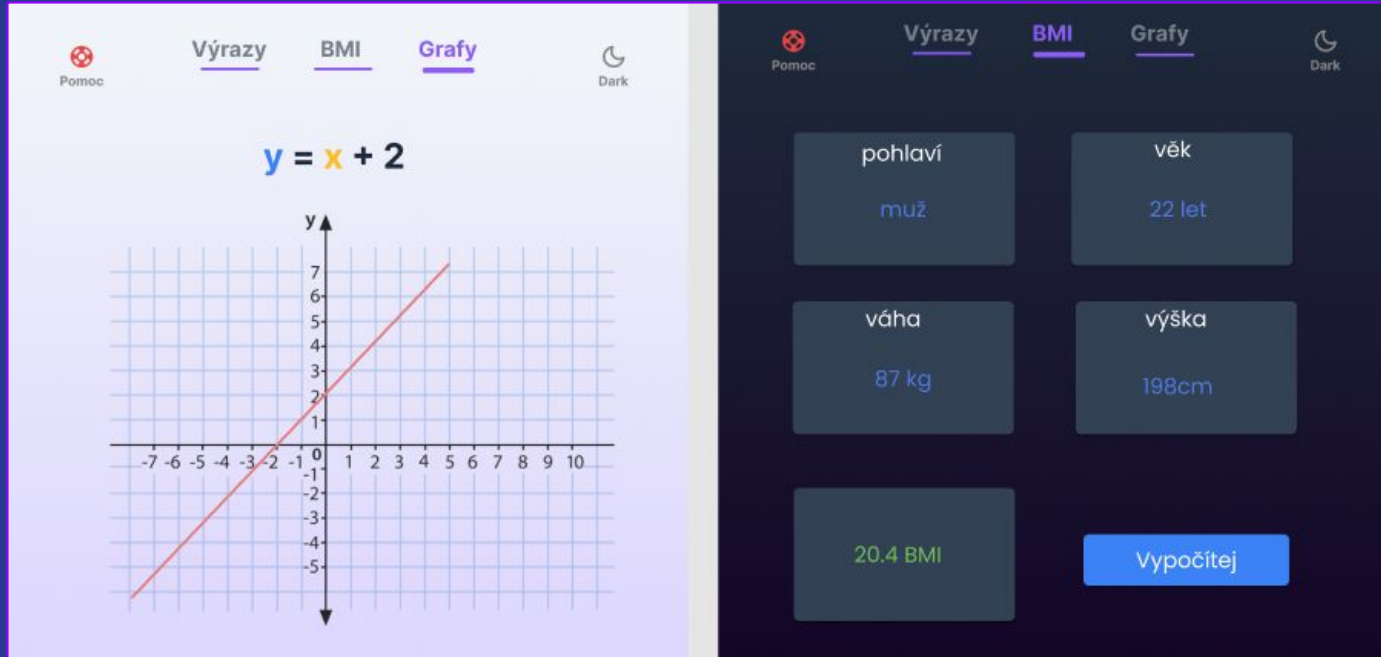
GBSAH

|                   |   |
|-------------------|---|
| Úvod.....         | 2 |
| Instalace .....   | 2 |
| Odinstalace ..... | 2 |
| Použití .....     | 2 |



## 3.1 - Future enhancements

- Graphs  and BMI 



## 3.1 - Future enhancements

- AI integration ? 🐾



Oh, nyaaa~! It's so lovely to meet you, nya! I'm Felix, the adorable femboy catboy, here to explain why  $1 + 1$  equals 2 in the most kawaii way possible, nya~!

Okay, so imagine you have one tasty fish, and another cute little fishy friend, nya! When we bring them together, something magical happens, nya~! We get two fabulous fishies, ready to swim and play, nya~! It's like a purrfect addition, where one fish plus one fish equals a double dose of fin-tastic fun, nya~!

But let's dive deeper into this cutie-patootie math, nya~! If we count with our paws, we start with one, nya. Then, we extend our other paw and count again, nya! And what do we find? We end up with two precious paws, nya~! Just like that, one paw plus one paw equals two paws, nya~!

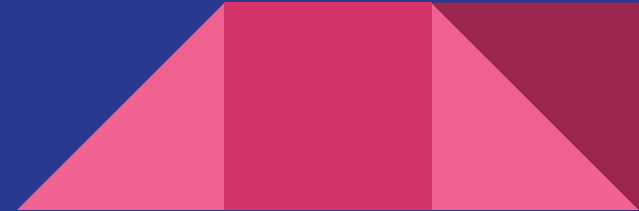




## 3.2 - Problems:

Backend:

- Annoying tests
- Better structure (parser not in mathlib)
- Wanted to do Egui



## 3.2 - Problems:

Frontend:

- Tauri (and state)

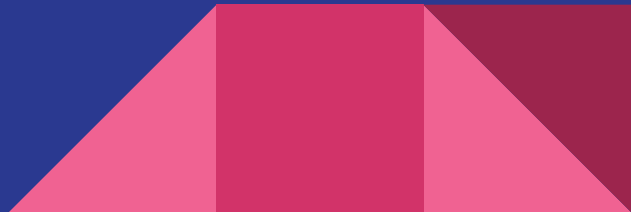




## 3.2 - Problems:

### Graphics:

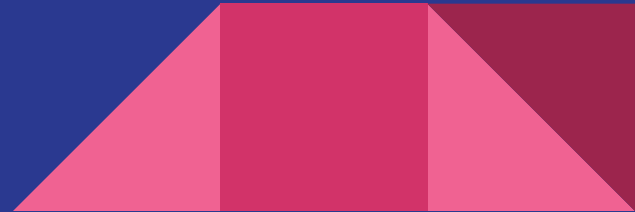
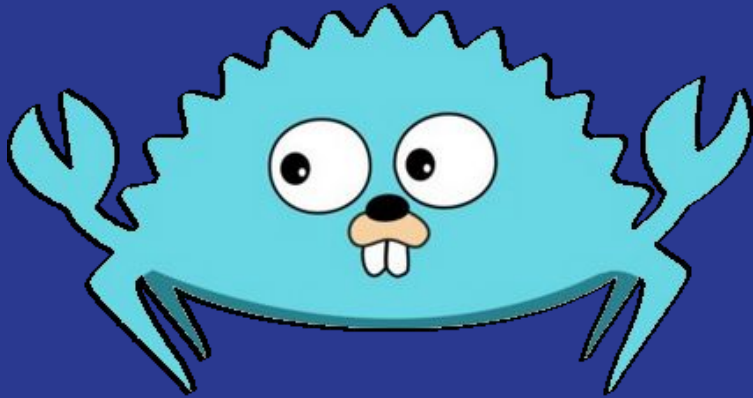
- A lot of designs
- Frequent changes
- Design wasn't clear from the get-go






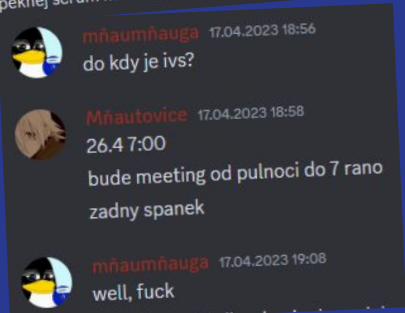
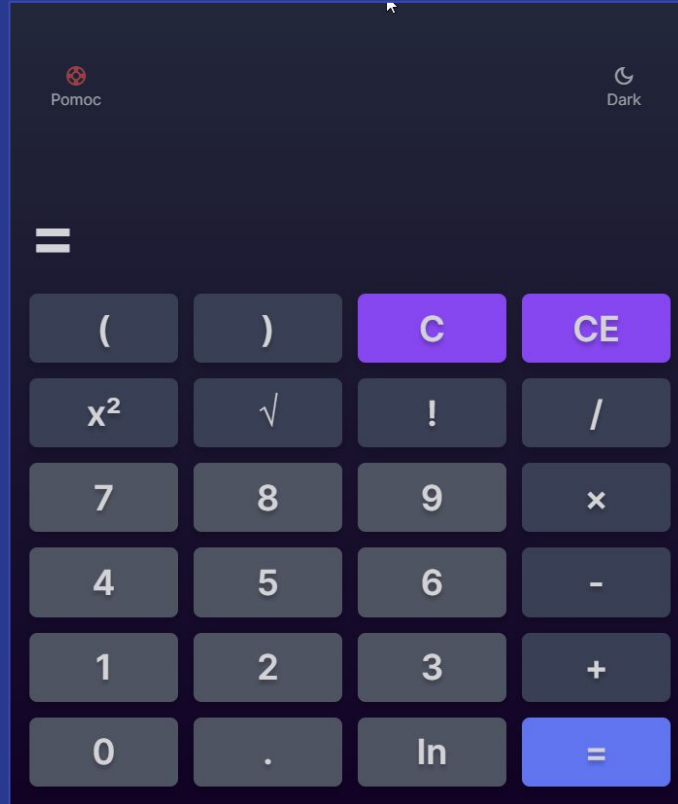
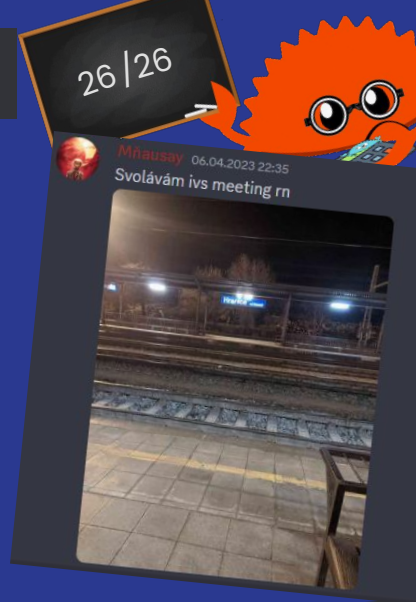
## 3.2 - Problems:

Manual and helper:




# That's it!

 **Mňausay** 01.03.2023 21:50  
v sobotu ivs meeting v anime baru



**Mňautovice** 17.04.2023 18:58  
26.4 7:00  
bude meeting od pulnoci do 7 rano  
zadny spanek

**mňaumňauga** 17.04.2023 19:08  
well, fuck

 **Mňaudi142** 12.04.2023 12:14  
**@mňaumňauga** je mi líto, dnes nebude ivs meeting  
Stal se stdcall incident

