

FUNKCIJSKI ZNANSTVENI KALKULATOR

Andrej Korica, Dominik Ružić, Krešimir Špehar



Uvod

- Znanstveni kalkulator u ovom projektu implementiran je pomoću Haskell-a te prikazuje snagu funkcijskog programiranja
- Sadrži grafičko korisničko sučelje
- Podržava različite matematičke operacije poput aritmetike, potencije i logaritama itd.

Izgled aplikacije



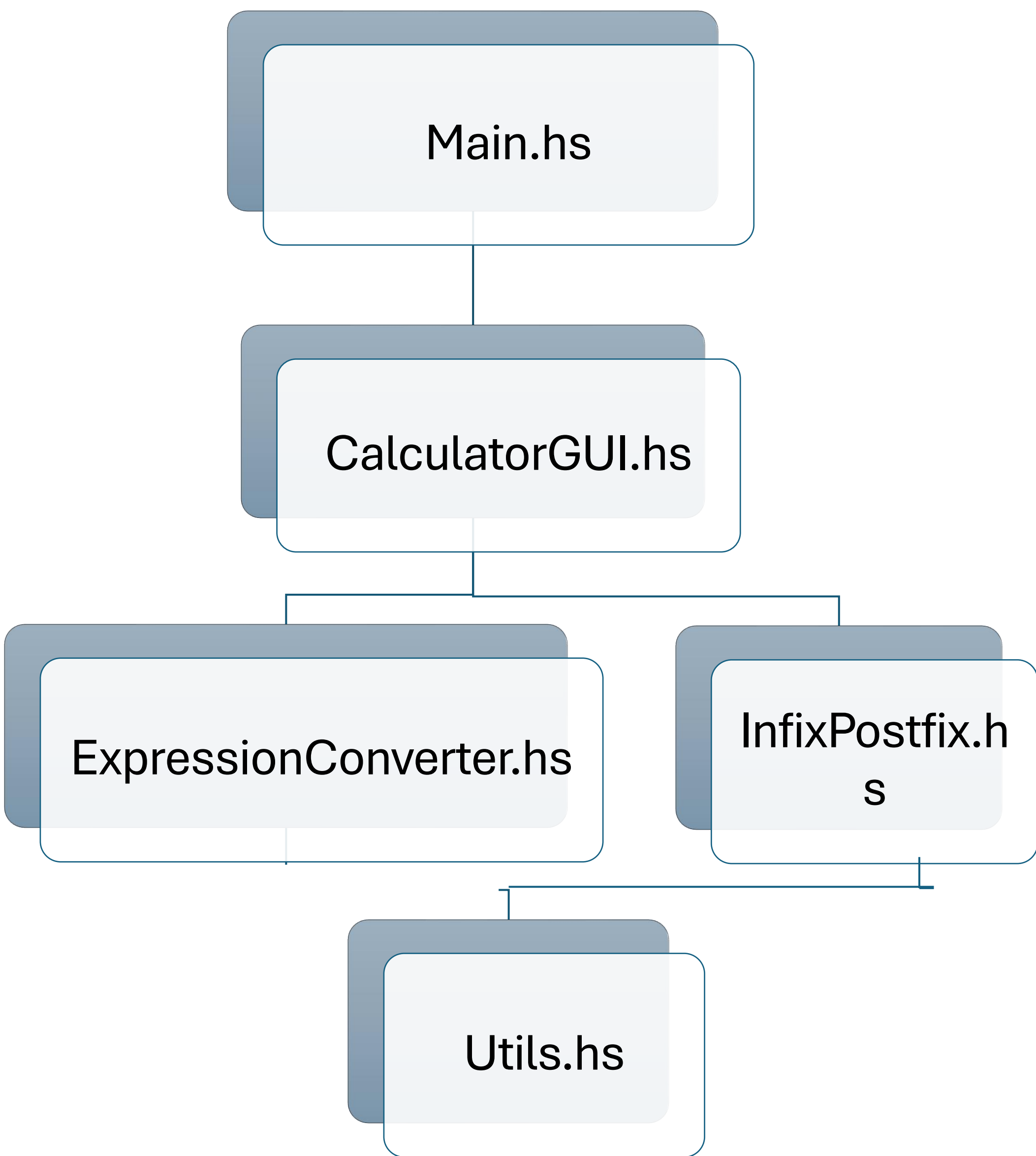
Ključne funkcionalnosti

- Čiste funkcije - nemaju nuspojava i uvijek proizvode isti učinak za isti unos
- Funkcije višeg reda - map, filter itd., intenzivno se koriste za obradu lista i drugih struktura podataka
- Usklađivanje uzoraka – koristi se za destrukuiranje podataka i eksplicitno rukovanje različitim slučajevima

Zaključak

- Haskell demonstrira snagu funkcijskog programiranja u implementaciji znanstvenog kalkulatora
- Korištenje čistih funkcija, funkcija višeg reda i usklađivanja uzoraka dovodi do robusnog, održivog i učinkovitog koda

Arhitektura



Prednosti

- Nepromjenjivost - jednom definirane varijable se ne mijenjaju, eliminira nuspojave i čini kod lakšim za otklanjanje pogrešaka
- Prvorazredne funkcije - tretiraju se kao prve klase dopuštajući im da budu proslijeđene kao argumenti, vraćene iz drugih funkcija i dodijeljene varijablama
- Konkurencija i paralelizam - čiste funkcije olakšavaju pisanje istodobnih i paralelnih programa jer nema nuspojava o kojima treba brinuti
- Modularnost - fun. Programiranje potiče rastavljanje problema na manje funkcije koje se mogu ponovno koristiti, s time poboljšava čitljivost koda i lakoću održavanja
- Analogno subklasiranje – dodavanje novih funkcija uniformnim procesom

Reference

Hutton, G. (2016). *Programming in Haskell*. Cambridge University Press.
Thompson, S. (1996). *Haskell: The craft of Functional Programming*.
Hudak, Paul (1989). *Conception, evolution, and application of functional programming languages*.