

Gymnázium, Praha 6, Arabská 14
obor programování



Habit Tracker

Ročníková práce

Rory Beneš 4.E
Duben 2024

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Praze dne 2.4.2024

Rory Beneš

1. Anotace

Cílem této práce je naprogramovat mnohoplatformovou aplikaci Habit Tracker. V této dokumentaci práce bude vysvětleno fungování a princip programu jako takového, popsán můj postup k jeho vypracování a vysvětlení nejdůležitějších procesů.

2. Obsah

3.	Charakteristika aplikace	1
4.	Použité technologie.....	1
5.	Aplikace	2
5.1	Návyky	2
5.2	Grafické zpracování	3
5.2.1	Grafy	3
6.	Finální zhodnocení.....	5
7.	Zdroje	5

3. Charakteristika aplikace

Habit Tracker je aplikace, ve které si uživatel zaznamenává své návyky, nastavuje cíle, kterých chce dosáhnout a sleduje své pokroky.

Uživatelé mohou sledovat své statistiky prostřednictvím grafů, které zobrazují, jak často a jak dlouho dosahují svých denních, týdenních nebo jiných cílů. Při každém novém záznamu dat se uživateli před očima graf záznamů změní a umožní instantní zpětnou vazbu.

Každému návyku má uživatel možnost nastavit vlastní cíl a periodicitu (Mým cílem je uběhnout každý týden 30km, nebo za noc naspát 8 hodin), tento cíl se pak také zobrazí do grafického zpracování a poskytne uživateli příležitost poznat, jak často svých cílů skutečně dosahuje.

Graf každého návyku je vygenerován zvlášť s vlastní podobou.

4. Použité technologie

Toto byl můj první projekt, který jsem zhotovil pomocí React Native – open-source UI software frameworku vytvořený Meta Platformami Inc, který spojuje nativní vývoj s javascriptovou knihovnou React, sloužící ke stavbě uživatelských rozhraní. Technologie React Native tedy umožňuje současný vývoj jak webových, tak i mobilních aplikací pro Android i iOS.

K vykreslování grafů jsem využil Victory Native – knihovnu Reactových komponentů určených pro interaktivní zpráhlednění dat. Všechny komponenty poskytované Victory lze použít k vizualizaci dat v různých formátech a podporují „kompletní“ přizpůsobení z hlediska stylů a chování, ačkoliv jsem se s nimi občas trochu omezovaný cítil.

5. Aplikace

5.1 Návyky

Návyky jsou základním stavebním kamenem celé aplikace Habit Tracker. Každý zaznamenaný návyk představuje klíčový prvek, který uživatelé sledují a analyzují, aby dosáhli svých osobních cílů. Návyky zabírají prakticky celou vizuální stránku aplikace, při prvním otevření vypadá domovská obrazovka prázdně, uživatel je vyzíván vytvořit první návyk, kterým prostor vyplní.

Při tvorbě nového návyku nezadáva uživatel jen název, ale i svůj osobní cíl. Nastavení konkrétního cíle totiž jedinci poskytuje jasný směr, měřitelný pokrok a motivaci k tomu, aby pracoval na změně. Ve své podstatě bude aplikaci používat ten, který chce ve svém životě něco změnit, v něčem se zlepšit, pravidelně něco trénovat, či naučit se něco nového. Takovému člověku konkrétní cíl a vizualizace jeho případného dosažení/nedosažení ohromně prospěje.

Konkrétní cíl si uživatel nastavuje v tomto případě jako cíl přesně měřitelný, což znamená, že je schopný jednoznačně určit, kdy ho dosáhl. Tato měřitelnost právě umožňuje sledovat veškerý pokrok a poskytuje uživateli jasnou zpětnou vazbu.

Údaje osobního cílu, kterými jsou *1) kolik*, *2) čeho* a *3) za jak dlouho* se ukládají a jsou spravovány třídou *Data*, která se dále stará o přidávání a čtení dat z lokálního úložiště.



Uložením nového návyku je uživatel přesměrován na domovskou stránku, která se pro něj nyní stává seznamem jeho návyků, kde si může kterýkoliv z nich rozkliknout, aby přidal data a zobrazil graf. Návyků může být vytvořeno neomezené množství.

5.2 Grafické zpracování

Grafické zpracování a obecný vzhled aplikace je velmi minimalistický. Nechtěl jsem, aby si aplikace hrála na něco většího a důležitějšího, než doopravdy je. Koncept Habit Trackeru je jednoduchý, doufám tedy, že vizuál působí podobným dojmem.

Barevností aplikace dost šetří. Barevností jsou zvýrazněny důležité prvky, například přidávání nového návyku, mazání existujícího, či cílové linie.

5.2.1 Grafy

Nejdůležitější součástí programu jsou samotné grafy záznamů každého návyku. Celý princip toho, že je jedinci prospěšná vizuální zpětná vazba stojí na těchto grafech. Vizuální zpracování totiž umožňuje pokrok vidět hned na první pohled, zpětnou vazbu poskytují nejjednodušeji jak to jde.

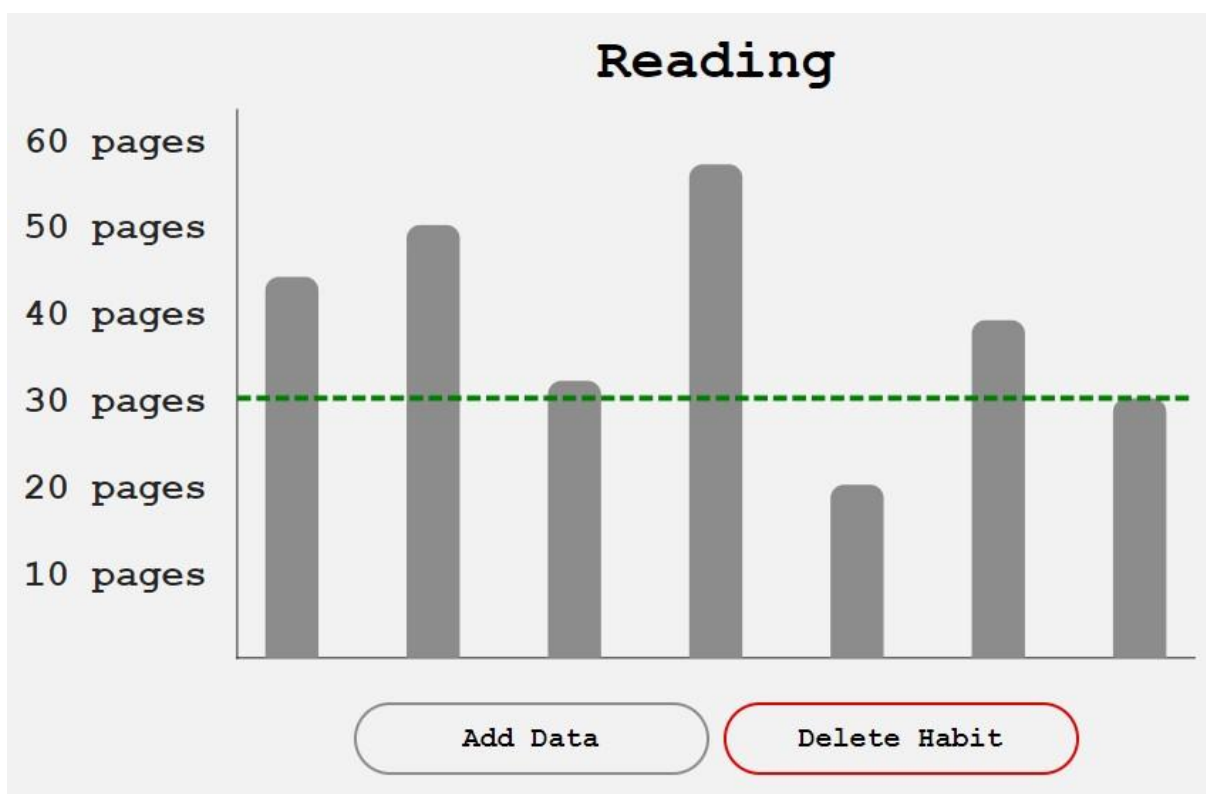
Uživatel může sledovat, jak se jeho návyky vyvíjí a mění v průběhu času. Tento vizuální přehled může být velmi motivující a může u uživatele posilovat odhodlanost pokračovat ve svém úsilí. Každému je příjemné, když ho někdo chválí za něco do čeho vložil čas a práci a to v jakémkoliv měřítku, stačí i pouhý diagram barevných sloupců protínající cílovou linii jakou nabízí například Habit Tracker. Dodává dále jistou strukturu jedincově úsilí a může ho donutit k pocítění větší zodpovědnosti za své zlovyky, kterými jsou například prokrastinace nebo nepravidelnost.

Graf vzniká hned ve chvíli, kdy uživatel přidává nový návyk do seznamu na domovské stránce. Když uživatel rozklikne panel návyku, uvidí prázdný graf čekající na datový input.

Při renderování grafu a přidávání dat do grafu je také implementovaná jednoduchá animace, která uživateli zpřehledňuje a zpříjemňuje přijetí vizuální zpětné vazby.

Jako typ grafu jsem pro tuto aplikaci zvolil svislý sloupcový, tedy takový, který znázorňuje složení sledovaného souboru pomocí obdélníkových pruhů, jejichž délka proporcionálně odpovídá velikosti hodnot, které znázorňují. Je to perfektní typ diagramu pro znázorňování toho, co potřebuji, tedy kolik čeho uživatel dokázal v nějakém určitém časovém rozmezí (den/týden/měsíc) pomocí jednotlivých sloupců. Dále v grafu hraje velkou roli vodorovná cílová čára, položené ve výšce hodnoty stanoveného cíle, kterou sloupce protnou pouze v případě jeho dosažení.

Souřadnicové osy ukazují hodnoty vygenerované na základně nastaveného cíle uživatele. Tedy například u návyku s cílem *spát 8 hodin denně* (název: *spánek*, kolik: 8, čeho: *hodin*, za jak dlouho: *za den*) se na y-ové souřadnicové ose objeví řada vzestupných čísel znázorňujících počet naspaných hodin. Na x-ovou osu se po prvním přidání dat objeví první sloupec. Po sedmi záznamech pak na ose lze zpozorovat průběh celého týdne.



Postup pro zadávání nových dat do grafu je přímočarý a velice jednoduchý. Po stisknutí tlačítka “Add data” se uživateli objeví dialogový box, do kterého stačí napsat číslo, není potřeba udávat čeho, jelikož tento údaj byl určen již během tvorby návyku a zůstává neměnný. Po přidání se hned do grafu přičte zadaná hodnota ke “dnešnímu” dni a vykreslí ve formě vyššího sloupce.

6. Finální zhodnocení

S odevzdanou prací jsem spokojen. Je jednoduchá a srozumitelná. Nabral jsem opět nové zkušenosti s novými nástroji světa programování.

7. Zdroje

Čerpal jsem primárně z dokumentací jednotlivých knihoven.