

Dokumentace

Tomáš Kolařík

Cíl projektu

Vytvořit aplikaci v javascriptu, která po načtení vstupního dokumentu ve formátu graphml zobrazí jeho obsah. Primárně má sloužit jako vizualizace letů v USA, což je také obsah přiloženého dokumentu v graphml. Letiště jsou vrcholy grafu a lety jsou hrany. V grafu bude možnost vybrat konkrétní letiště, kde se poté zobrazí lety z něj. Letišť je možné vybrat více. Obecně lze vybrat region zájmu grafu.

Postup

Nejdříve jsem se rozhodl, jakou technologii použít na práci s grafy. Vybíral jsem mezi tím ručně graf kreslit jako SVG, nebo použít framework pro práci s grafy. Rozhodl jsem se použít framework, konkrétně viz.js. Ten má své připravené kontejnery pro ukládání vrcholů a hran a jejich vykreslování do canvasu.

Po vybrání frameworku bylo potřeba načíst data ze souboru graphml, což je soubor formátu XML. Pomocí knihovny xml2json.js jsem si ho načel do jsonu, kde už jsem iterací načel vrcholy s patřičnými parametry a jejich hrany.

Po nastavení základního vykreslování bylo potřeba zajistit, aby hrany byly skryté a zobrazili se až po vybrání zdrojového vrcholu. To jsem udělal pomocí skupiny, které framework viz.js podporuje. Po vybrání vrcholu ho přiřadím do skupiny „vybraní“, které určuje jeho barvu a podobně a všechny jeho edge zobrazím. Jelikož jsem při každém kliknutí musel projít všechny hrany a po jednom jim nastavit aby se zobrazovali, byla aplikace velmi pomalá. Vytvořil jsem si tedy při načítání mapu, která říká, jaký vrchol patří jakému vrcholu a nemusel tak při každém vybrání vrcholu všechny hrany. Aplikace však byla stále pomalá. V dokumentaci ke frameworku jsem našel možnost nastavit propagaci změn hromadně, jako jednu velkou transakci a ne pro každý vrchol/hranu zvlášť. Po této optimalizaci již aplikace běžela přijatelnou rychlostí.

Poté jsem implementoval možnost posouvat vrcholy. Framework nabízí událost dragStart a dragEnd na vrcholech. Když dojde k události dragStart, uložím si souřadnice vrcholu. Když dojde k události dragEnd, uložím si do pole objekt popisující odkud kam byl vrchol přetažen. Ten má zároveň funkci pro update tohoto vrcholu a také pro navrácení této změny. Když pak uživatel mačká tlačítko zpět, popují se z pole tyto objekty a volá se na nich funkce vracející provedenou změnu.

Na závěr jsem implementoval možnost zobrazení vlastního nahraného graphml souboru pomocí File api. Problém byl, že formát graphml je poměrně vágní a graphml soubory tak mohou mít jinou vnitřní strukturu xml. Přidal jsem proto jednoduchou kontrolu toho, zda je vnitřní formát takový, jaký očekávám. Načtu soubor graphml a převedu ho na json a zkusím, jestli tam, kde očekávám, že budou hrany a vrcholy, něco je. Pokud se soubor povede převést (tzn. je formátu xml) a obsahuje na očekávaném místě informace o hranách nebo vrcholech, vykreslím ho. Jinak vypíšu chybu formátu.

Detailní popis je ve zdrojovém kódu.

Popis funkčnosti

Aplikace po otevření zobrazí graf načtený ze souboru graphml s pozadím mapy USA. V tomto grafu lze vybírat region zájmu. Lze vybrat libovolný počet vrcholů. Po vybrání vrcholu se zobrazí hrany,

které z něj vedou. Vrcholy je možné také posouvat a tyto změny poté vracet zpět. Aplikace také umožňuje nahrát vlastní soubor, který poté již bez pozadí zobrazí. Musí ale mít stejný formát a vnitřní strukturu jako původní soubor graphml, což aplikace kontroluje a případně vypíše chybu formátu.