

Úvod

Účel produktu

Tento produkt slouží pro demonstrační účely na předmět MTI/STIN jako semestrální projekt. Projekt má ukázat, že studentovy znalosti jsou dostatečné pro získání zápočtu. Semestrální projekt by měl ukázat, že student umí napsat DSP dokument (neboli tento dokument), umí si dobře vybrat programovací jazyk, rozmyslet si, jak správně naprogramovat i takto malý projekt. Mezi další požadavky ke zdárnému získání zápočtu je potřeba umět napsat testy (unit i integrační). Nakonec je potřeba umět automaticky spustit aplikaci na studentem vybraném serveru.

Rozsah produktu

Produkt není nijak složitý nýbrž jednoduchý. Používá jednoduché funkce. Hlavní náplní je informovat uživatele o základních informacích.

Definice a zkratky

Klient – část programu zobrazující požadované funkce programu a součást programu komunikující se serverem

Server – část programu uchovávající data nebo zjišťující data z jiných zdrojů

Operační systém – program, který dává života schopnost zařízením a stará se o bezstarostný běh dalších programů a aplikací

Hosting – počítač nebo server na kterém běží naše aplikace

DSP – dokument, který čtete

Metoda POST – složí pro zasílání informací mezi klientem a serverem

Metoda GET – slouží k získání webové stránky, pro získání vizualizace klienta a dalším podobným požadavkům

Webová stránka (aplikace) – interaktivní obsah získaný na internetu

HTML – vizualizace webové stránky, informace o tom, co se má zobrazit na webové stránce

Implementovat – napsat/naprogramovat řešení nějakého požadovaného problému

Knihovna – skupina funkcí řešící skupinu daných problémů

Funkce – část kódu řešící jeden problém

URL – jednoznačný odkaz na nějakou požadovanou informaci, funkci nebo soubor

Bot – zařízení nebo webová stránka napodobující nebo zastupující člověka

Systémový přehled

Serverová část bude běžet na operačním systému Linux. Klientská část nemá žádné požadavky na operační systém, protože počítám s během ve webovém prohlížeči.

Reference

[Odkaz na use case diagram.](#)

Celkový popis

Perspektiva projektu

Perspektiva úspěchu projektu na dnešním trhu je celkem malá, protože existuje už hodně implementací od různých firem, ale pro účely semestrální práce je toto zadání dostačující, protože lze i lehce rozšířit v případě nových požadavků.

Funkce produktu

Mezi základní funkce produktu patří komunikace mezi klientskou částí a serverovou částí, které jsou každé na jiném hostingu. Přihlašování není třeba a mělo by se myslet na ochranu před DDOS útoky. Musí jít o otestovaný kód. Je potřeba otestovat kód unit i integračními testy. Je třeba zajistit automatické nahrávání na server po úspěšných testech. Testy by měli být exportované do externího nástroje. Na programovacím jazyce nezáleží. Je potřeba myslet na to, že může přijít dotaz na to, co náš server neumí obsloužit, proto musíme ošetřit i tyto stavy.

Klient by měl umět zobrazovat informace ve formě HTML. Zobrazovaný jazyk by měl být jen jeden vybraný programátorem. Já jsem si vybral anglický jazyk. Klient by měl přijímat dotazy ve formátu takovém, že první část textu ho nezajímá a až nalezne slovo jak/jaká/jaké/jaký, tak zpracuje příkaz za oddělovačem. Vzory příkazů budou předem dány v nápovědě. Klient volá funkce serveru.

Server by měl vracet nápovědu, čas na serveru v okamžiku dotazu, jméno serveru a kurz v okamžiku dotazu (prozatím jen z České koruny na Euro). Poté by měl dokázat vrátit rozšiřující funkci starající se o výpis historie kurzu měny (od startu aplikace), kterou jsme si přidali do požadavků na cvičení, kdy jsme probírali DSP dokumenty a upřesňovali informace. Není potřeba uchovávat historii dotazů a vrácených odpovědí.

Charakteristiky uživatele

Uživatel může být prakticky kdokoliv, protože v požadavcích je požadováno, že řešení nesmí obsahovat přihlašování, proto logicky si může vyzkoušet aplikaci kdokoliv, kdo má odkaz, proto musejí být dobře ošetřené vstupy od uživatele.

Propojení

Propojení klienta a serveru bude realizováno pomocí metod POST a při příchodu na server metodou GET. Metodou GET na začátku zašlu HTML stránku pro vizualizaci klienta aplikace. Metodou POST budu zasílat informace o tom, jaký dotaz si uživatel vybral.

Předpoklady a rozšíření

Předpokládám hlavně rozšíření u funkce pro zobrazení aktuálního kurzu. Hlavně očekávám rozšíření na více měn a také rozšíření funkce historie kurzů, kdy bude moct být začátek sledování i před startem aplikace. Očekávám také přidání úplně nové funkce serveru (např. Jaký je datum?).

Systémové požadavky

Externí rozhraní

Jedno z použitých externích rozhraní bude knihovna, která implementuje funkci serveru (např. u Node.js Express). Použiji také knihovnu pro ochranu napadení aplikace přes HTML (např. u Node.js Nunjucks) a další knihovny pro zjištění datumu a času v daný okamžik.

Funkce

Když bude server obsluhovat požadavek na aktuální kurz budu očekávat že webová stránka s URL: <https://www.cnb.cz/cs/financni-trhy/devizovy-trh/kurzy-devizoveho-trhu/kurzy-devizoveho-trhu/vybrane.html?od=01.02.2022&do=31.03.2022&mena=EUR&format=html> bude fungovat. Pro ukázkou jsem přidal odkaz na kurz z České koruny na Euro od 1. února 2022 do 31. března 2022. V případě zjišťování kurzu v daný den použiji týdenní výpis neboli 7 dní (i s aktuálním datem) a vrátím kurz na posledním řádku s datem na tomto řádku. Sedm dní budu vracet abych měl jistotu návratu aktuálního kurzu – hlavně kvůli víkendům a svátkům.

Požadavky na výkon

U této aplikaci nebyli vysloveni požadavky na výkon stroje, kde to poběží z důvodu velikosti projektu, protože je projekt malý.

Databázové požadavky

Databázové požadavky nejsou žádné. Zadavatel poukázal na to, aby programátor zkusil vytvořit aplikaci bez databáze.

Vzhledové propojení

Vzhled by měl být pěkný s prvky příkazové řádky nebo inteligentního bota.