



VNL1

Lokaskýrsla

Maí, 2019

Nafn nemanda: Guðmundur Ingi Jónsson

Kennitala: 170696-2559

Umsjón: Arnar Ingi Gunnarsson

Inngangur

Í þessari skýrslu verður farið yfir allt sem gekk vel, allt sem gekk illa og allt sem var breytt. Oft er slík skýrsla kölluð retrospective eða lokaframvinduskýrsla. Lokaframvinduskýrslur eru mikilvægur partur af allri hugbúnaðargerð. Með því að skoða velgengni verkefnis er hægt að fá betri sýn á skilvirkni hóps í verkefnavinnu. Þar sem allir fá tækifæri að koma sinni skoðun á framfæri er hægt að horfa gagnrýnari augum á hópinn en áður og með samvinnu er hægt að vinna úr þeim góðu og slæmu atriðum til að vinna í framtíðinni gangi betur.

Einnig fæ ég tækifæri til að lýsa breytingum, erfiðleikum og velgengni við vinnu mína í þessu verkefni. Til dæmis þá vill ég koma því á framfæri að það getur verið að einhverjir hlutar fóru framhjá mér í útfærslu forritsins eða við vinnslu þeirra skýrslna sem ég gerði. Það geta líka verið góðir eða slæmir hlutir sem fengu ekki gagnrýni annara og það veldur því að ég sé þá ekki heldur.

Eftir fyrsta skiptið þar sem ég gerði svona skýrslu þá hef ég verið að hugsa um þá áfanga í skólanum (viðskiptafræði) sem myndu gagnast af svona verkefni í stað jafningjamats. Þar sem jafningjamat er nafnlaust og trúnaðarmál gefst ekki séns fyrir aðra aðila til að heyra það sem aðrir hafa að segja eða útskýra sitt mál og það veldur því að þeir aðilar gera sér kannski ekki grein fyrir því sem gert er vitlaust nái ekki að bæta sig í framtíðinni. En framvinduskýrsla getur bætt samvinnu ef henni er beitt rétt.

Greining á hvað fór vel og hvað úrskeiðis við útfærslu verkefnis

Flest allt gekk vel í gerð forritsins. Það hjálpaði mikið að vera búinn að gera svipað verkefni tvisvar sinnum áður. Þá í verklega námskeiði 1 í árslok 2017 og í lokaverkefni í gagnaskipan. Einnig er hentugt að vera kominn lengra í tölvunarfræði náminu en hefðbundnir nemendur VNL1 þar sem auðveldara var að útfæra ýmsa hluti. Verklega námskeiðið í árslok 2017 fór fram í C++ þannig ekki var hægt að hafa það til hliðsjónar en verkefnið sem gert var í Gagnaskipan var gert í Python þannig hægt var að byggja ofan á það sem nú þegar var komið. Af þeirri virkni var lestur gagna úr txt skjölum sem gögn voru aðskilin með semikommu, eina sem þurfti að breyta þar var nafnið á skalinu og skráarendingin (.txt -> .csv) og nöfn breyta sem tilheyrir gagnaskránni. Einnig var til staðar virkinn til að bæta við línu í skjal og takmörkuð virkni fyrir eyðingu úr skrá. Eftir það voru ýmsir hlutir breyttir og útfærðir til að þjóna ýmsum tilgangi.

Ekki var mikið um vesen sem betur fer en þau vandamál sem kom upp voru erfið og ekki náðist að leysa þau öll. Þýðing gagna úr og í skjöl voru mismunandi eftir því hvort „utf-8-sig“ eða „utf-8“ virkaði. Þetta kom ekki upp í byrjun þar sem verkefni sem leyst var áður í öðrum áfanga var búið að leysa það. Ef ekki er lesið inn skjöl með „utf-8“ kemur upp spurningamerki á lettri eins og íslenskum stöfum. Það sem kom upp við útfærslu á kerfinu var að þegar skjal var uppfært kom upp unicode merki í línu 3. Ekki fékkst ástæða fyrir þeirri hegðun en það náðist að laga það með því að breyta kóðuninni á skjalinu.

Það að tengja klasa á milli laga gekk vel í byrjun þangað til það uppgötvaðist að nokkur tilvik voru stofnuð af bæði application laginu og gagna laginu sem gerði það að verkum að ef eitt tilvik las inn og annað skrifaði út þá skrifaði það bara út því sem breytt var og eyddi öllu hinu út. Ástæðan fyrir stofnun auka tilvika var vegna þess að getið var í ákveðið lag efst í öðru lagi og ekki gekk að flytja það á milli falla, þannig getið var aftur í sama lag án þess að vita að það stofnaði annað tilvik. Lausn á því vandamáli kom við stofnun „self“ breytu í „__init__()“ fallinu

Þegar bætt var við virkni sem gerir notanda kleift að stofna nýja línu í skjal uppgötvaðist það að ekki var verið að halda utan um einkennisnúmer þeirra lína sem eru í skjalinu. Þar sem notandi á ekki að þurfa að slá inn einkennisnúmer nýrra lína þurfti að bæta við þeirri virkni. Það var gert með því að hafa hæsta númer sem lesið var inn úr skrá vera

einkennisnúmer nýrra lína. Með þeirri virkni var hægt að halda utan um nákvæma einkennistölu jafnvel þótt að tilheyrandi línu yrði eytt þá myndi næsta nýja lína fá það númer, vegna þess að við breytingar á skrá mun allt lesast inn upp á nýtt ásamt hæstu einkennistölu.

Eftir að kerfi var að mestu leyti full klárað var skjölum skipt upp í möppur. Þegar það var gert þurfti að skrifa allar tilvísanir í önnur skjöl upp á nýtt. Gagnaskjölum (csv) var léttilega breytt þar sem einu breytingarnar sem voru nauðsynlegar voru í nafninu á skjalinu. Hins vegar kom upp vandamál við breytingu á „import“ skilgreiningum þar sem ekki var hægt að vitna beint í skjölin í sömu möppu og python vildi ekki skilja leiðarvísin í réttu skjölin. Eftir langa leit á Google var prófað að bæta við auka kóða fyrir ofan „import“ skipunina og það virkaði. Líklega vegna þess að þessi auka kóði hjálpaði við að skilgreina nákvæmlega staðsetninguna á skjalinu. Einnig ætti það að virka á mismunandi tölum.

Bætt var við innskráningarsíðu fyrir notendaviðmótið þar sem þeir notendur sem greindir voru hafa mismikinn aðgang að kerfinu. Vegna þess fyrirnefnda vandamáli við tengingar á skjölum gekk ekki vel að búa til sér skjal sem vitnar í notendaviðmótslag kerfisins þannig innskráningarvirkninni var bætt við neðst í notendaviðmótkóðanum

Þegar listi af gögnum úr gagnasöfnum var birtur var allt flokkað eftir fyrsta númerinu í einkennisnúmerinu, og svo öðru og koll af kolli. Sem sagt ekki í réttri röð. Það sem var að valda því var að þegar gögnin voru lesin úr skjalinu þá notuðust þau við einkennisnúmerin í skjalinu til að raða í sorted dict en þar sem þau voru strengir þá var þeim raðað á rangan hátt. Það var lagað með því að notast við self._id breytturnar sem voru til staðar sem túlkuðu þau sömu einkennisnúmer sem alvöru tölur í stað stengs. Eftir það þá birtust listarnir í réttri röð

Greining á hvaða kröfur voru útfærðar, hverjar ekki, hverjum var breytt og afhverju

Allar kröfur voru útfærðar en ekki endilega nákvæmlega þær sem settar voru í þarfagreiningarskýrslunni. Af þeim sem héldust óbreyttar voru:

- Stofna leitarhóp
- Setja saman tæki og meðlimi í leitarhóp
- Bæta við tækjum í kerfi
- Bæta við meðlimum í kerfið
- Sækja upplýsingar um staka meðlimi
- Sækja upplýsingar um stök tæki
- Breyta upplýsingum um staka meðlimi
- Breyta upplýsingum um stök tæki
- Eyða upplýsingum um staka meðlimi
- Eyða upplýsingum um stök tæki
- Notendaviðmót í texta
- Tími útkalls
- Ljúka útkalli

Þessar kröfur voru flestar svipaðar hvor annarri og oft var bara afritað kóða og breytt og ef ekki þá var útfærsla minniháttar. Fyrir utan notendaviðmót í texta. Þá var byggt allt frá grunni en samt sem áður var notast við afritun.

- Ekki hægt að hafa sömu tæki og meðlimi í tveimur virkum leitarhópum

Sú virkni var útfærð síðast af öllum og þar sem önnur virkni var til staðar sem staðfestir að númer sem slegin eru inn eigi við önnur einkennisnúmer var hægt að bæta einu skrefi við það sem skoðar leitarhópa virkra útkalla. Hugsunin á bak við þessa virkni er leitarhópar eru stofnaðir með það í huga að svara útkalli og ef einstaklingur er í virku útkalli er ekki hægt að bæta honum við annan hóp sem á að fara í útkall.

Þær kröfur sem voru breyttar eru:

- Legacy skjal með upplýsingum um útköll
- Staðsetning leitarhóps

- Skoða virka leitarhópa

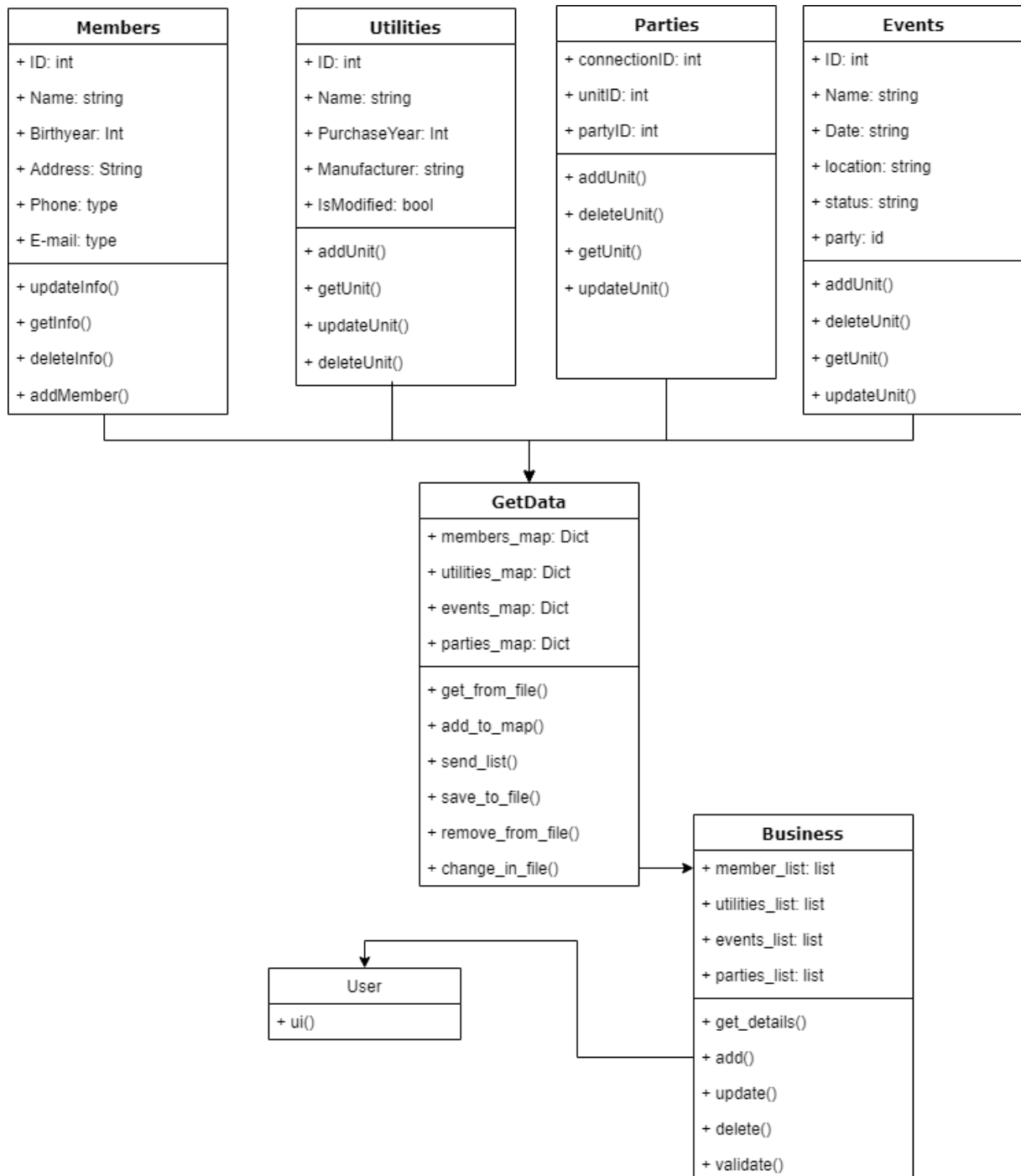
Legacy skjal átti að vera útfært en í stað þess var einum strengi bætt við í gagnalínu sem gefur til kynna hvort útkall sé virkt eða ekki. Með því að gera það þannig var hægt að hafa einungis eitt gagnaskjal með þeim upplýsingum.

Staðsetning leitarhóps var breytt í staðsetningu útkalls. Um er að ræða villu í orðalagi. Leitarhópar eiga að geta verið endurnýtanlegir og þess vegna er óþarfi fyrir þá að bera staðsetningu. Ef þörf er á því að finna staðsetningu leitarhóps er best að finna það virka útkall sem hann er í og þá sést staðsetning.

Og að lokum er krafan um að skoða virka leitarhópa. Eins og með staðsetningu var óþarfi fyrir leitarhóp að bera virkni þeirra þar sem þeir bera þá virkni miðað við útköll. Ef virkni eða staðsetning var í skjali leitarhópa væri nauðsynlegt að uppfæra margar línur í þeirra skjali og það er óþarfi.

Nokkrum kröfum var bætt við eins og sú virkni að notandi hefur ávallt möguleika á því að fara til baka í viðmótinu og staðfesting á að þau gildi sem slegin eru inn eru við hæfi annars birtast villuskilaboð. Einnig innskráningarsíða þar sem stjórnandi hefur meiri aðgang en aðrir notendur. Aðrar kröfur eru minniháttar en hjálpa samt sem áður forritinu í keyrslu.

Uppfært klasarit



Lokaorð

Lokaframvinduskýrsla sem þessi er mikilvæg til þess að hjálpa nemendum að sjá hvað gekk vel og hvað mætti betur fara í framtíðinni. Slík endurskoðun getur einnig nýst í atvinnulífinu óháð iðnaði. Endurgerð klasaritsins þjálfar mann einnig í hvernig þau eiga að líta út í upphafi.

Að mínu mati ættu þessi námskeið að hafa verkefnalýsingu eins og þessa sem er björgunarsveit. Þar sem ófullnægjandi útfærsla kröfu í kerfum eins og fyrir pizzastað eða bílaleigu geta valdið vandamálum hjá viðskiptavinum fyrir. Þá getur ófullnægjandi krafa í kerfum eins og fyrir neyðarlínu, slökkvilið eða sjúkralið haft mun verri áhrif og þá er meiri þrýstingur á nemendum að gera verkefnið vel.

Ég er þakklátur fyrir þann séns sem ég fékk til þess að taka upp þennan áfanga svo ég gæti útskrifast með samnemendum mínum.

Takk