

Ekkert mál. `searchHotel()` tekur inn frá `SearchView` lista af öllum mögulegum breytum sem hægt væri að leita eftir og skilar frá sér öðrum lista af `Hotel` hlutum sem uppfylla þessi skilyrði. Athugið að í verkefnalýsingu okkar bjóðum við upp á grunnleit og ítarlega leit; grunnleitin sendir bara inn í `searchHotel` breyturnar `startDate`, `endDate` og `numberOfGuests`, meðan ítarlega leitin sendir inn þessar sömu breytur og svo eitthvað óþekkt mengi af hinum. Ég læt fylgja með Sequence Diagram af notanda sem leitar að hóteli og pantar eftir útkomunni, ef það skyldi gera þetta skýrara.

Nokkrir aðrir punktar um `searchHotel()` sem gætu hjálpað:

- Við höfum ekki ákveðið hvernig nákvæmlega við látum fallið aðgreina hvaða stök í List breytunni "parameters" séu hvaða leitargildi. Einn möguleiki er að hafa þau alltaf í sömu röð, þannig að `parameters[1]` sé alltaf `startDate`, `parameters[2]` sé alltaf `endDate` og `parameters[3]` alltaf `numberOfGuests`; og ef `parameters` er lengra þá sé `parameters[4]` alltaf `priceRange`, `parameters[5]` alltaf `rating`, o.s.frv. Þetta er ekki fullkomnasta lausnin, en við ákváðum að hún væri skömminni skárri en milljón overloaded `searchHotel()` föll sem tækju hvert fyrir sig inn einhverja ólíka samsetningu af breytum.

- Ef notandinn valdi ekkert fyrir eitthvað ákveðið skilyrði (annað en grunnskilyrðin) þá mun viðkomandi `parameters[x]` sæti vera `null` eða eitthvað `null-equivalent` gldi. Það jafngildir því að við þurfum ekki að útiloka neitt byggt á `parameters[x]`.

- Sumar inntaksbreyturnar eru sjálfar fylki eða List. Við munum enda með skemmtilegar tilvísanir eins og `parameters[5].value[1]` eða eitthvað í þá áttina. Aftur, ekki fallett, en auðveldara en að vera með hundrað `searchHotel()` föll.

- Allar breytur sem eru fylki eða List eru "either-or", ekki "and". Ef notandinn valdi t.d. bæði 1-stjörnu og 2-stjörnu hótél í leit sinni (sem veldur því að `priceRange[0]==1` og `priceRange[1]==1`) þá mun `searchHotel` einungis útiloka þau hótél sem uppfylla ekki a.m.k. *eitt* leitarskilyrðanna fyrir þá breytu, en þau þurfa ekki að uppfylla bæði. Eins ef hann valdi `possibleAirports: { "Keflavík", "Akureyri" }` þá mun `searchHotel` skila öllum hótélum í Keflavík *og* öllum hótélum á Akureyri (og auðvitað takmarka svo listann eitthvað eftir öðrum leitarskilyrðum).

- Breytan `hotelFacilities[]` er boolean. Við verðum með fyrirfram skilgreindan lista yfir "Wifi", "Sundlaug", "Hótél bar", etc., og munum merkja við '1' í viðeigandi sæti í `hotelFacilities` ef notandinn vill hafa slíkt, en '0' annars.

- Breytan `possibleLocations[]` er int. Hún mun gilda um svæði á einhvern hátt '1' verður Vesturland, '2' verður Höfuðborgarsvæðið, '3' Suðurland, eitthvað þannig.

- Til að finna hvort hótél á laust nægilega mörg herbergi fyrir `startDate` og `endDate` tímabilið kallar `searchHotel()` á fallið `checkAvailability()` fyrir hvert `Hotel` instance. Það fall skoðar `freeRoomsPerDate` breytuna sína fyrir hvern dag milli `startDate` og `endDate` og skilar int x, þar sem x er fjöldi herbergja sem er laus *hvern einasta dag* á tímabilinu. Til dæmis ef `startDate` er 1. maí, `endDate` er 5. maí, og `checkAvailability` kemst að því að 1., 2., 3., 4. og 5. maí eru laus herbergi 10, 10, 3, 10, þá skilar það 3, því þú vilt ekkert fá upp hótél í leitinni þinni sem getur ekki tekið við öllum þeim sem þú ert að ferðast með út alla ferðina.

- `SearchView` fær til baka þennan lista af `Hotel` hlutum og sendir hann í `ResultsView` með fallinu `displayResults` (sem á að vera public, ekki private - það er óvart merkt með '-' frekar en '+' í UMLinu

okkar). ResultsView birtir svo listann af hótélum og leyfir þér að velja hvort þú vilt bóka eitthvað þeirra.

Ef þið hafið einhverjar fleiri spurningar, hafið bara samband :-)

--Hjalti