

Harjoitustyö

Harjoitustyönä oli tehdä Postin pakettiautomaatteja hyödyntävä postisovellus, jolla voi lähettää paketteja automaattista automaattiin. Toteutus tehtiin Javalla ja JavaFX:llä.

Ohjelman suunnittelu

Aloitimme ohjelman suunnittelun lukemalla tehtävänannon huolellisesti. Kasasimme tehtävänannon mallikäyttöliittymän pohjalta Scene Builderissa käyttöliittymän, johon aloimme suunnitella toiminnallisuutta Discordin välityksellä. Emme laatineet aluksi täydellistä suunnitelmaa, vaan palasimme aina silloin tällöin tehtävänantoon tarkastelemaan vaatimuksia. Suurin osa suunnittelusta tapahtui samaan aikaan koodauksen kanssa.

Suunnittelun apuna käytimme myös tehtävänannosta löytyvää luokkakaaviota.

Ohjelman teko

Koodasimme ohjelmaa omilla koneillamme ja kommunikoimme Discordin kautta. Suurimman osan ohjelmasta teimme samaan aikaan, eli pystyimme koko ajan keskustelemaan siitä mitä toinen oli tekemässä ja miten toisen tekemää toimintoa voi käyttää omassa koodissa. Myös virheiden eliminointi sujui hyvin, sillä toiselta sai hyviä vinkkejä niiden korjaukseen juuri sillä hetkellä.

Versionhallintaan käytimme Gitiä ja GitHubia. Versionhallinnan käyttö oli erittäin hyödyllistä ja saimme nopeasti uudet koodit koneelta koneelle.

Toiminnallisuudet

Perusvaatimukset:

- SmartPost-automaattien lisäys kartalle
 - o Jouduimme korjaamaan index.html:ää, sillä alun perin se ei piirtänyt kaikkia automaatteja kartalle. Vaihdoin Googlen hakemaan pisteet koordinaattien perusteella osoitteen sijaan, jolloin kaikki SmartPostit sijoittuvat kartalle (Akaalla neljä automaattia, mutta tehtävänannon index.html sijoitti kartalle vain kolme)
- Pakettien luonti varastoon
 - o Pakettien luonnissa virheentarkastusta ja virheistä ilmoittaminen käyttäjälle
- Pakettien lähetys ja reitin piirto
 - o Paketti voi mennä rikki kuljetuksen aikana ja lopputulos kirjoitetaan lokiin
- Kartan tyhjennys

Laajennetut vaatimukset:

- Loki, joka pitää kirjaa istunnossa tapahtuvista asioista (lähetetyt paketit, varastoon jääneet paketit yms.)
- Ohjelmaa suljettaessa loki kirjoitetaan log.txt-tiedostoon
- Käyttöliittymässä käytetään välilehtiä ja uusia ikkunoita
 - o Pääikkuna ja loki välilehdissä
 - o Pakettien lisäys ja luokkainfot uusissa ikkunoissa

Laajin vaatimustaso:

- Käyttöliittymä ei käytä pelkästään standardikomponentteja, vaan niitä on muokattu visuaalisesti
 - o Ulkoasu style.css-tiedostossa
 - o Kaikkien JavaFX-elementtien väriä ja tekstin ominaisuuksia muutettu
 - o Välilehtivalikon ulkoasua muokattu
 - o Ohjelmalla oma kuvake ja logo

- Virheentarkistus korostaa virheelliset kentät punaisella
- Paketeilla on omat hienot kuvat, kun ne lähetetään
 - Paketti kulkee reitin mukana kartalla
 - Suurin osa toteutuksesta index.html:ssä Google Maps JavaScript API:lla
 - Joukossa myös easter eggejä (.mp3-tiedosto toistetaan, kun oikea tuote lähetetään oikeaan paikkaan), joten koodi on ”piilotettu” (<http://www.javascriptobfuscator.com/Javascript-Obfuscator.aspx>)
 - Alkuperäinen koodi tiedostossa assets/easter_eggs.html

Työnjako

Ilari:

- Käyttöliittymän suunnittelu
- Käyttöliittymän kustomointi
- Pakettien ja tuotteiden käsittely
- Virheentarkistus
- Pakettien omat kuvat ja easter eggit

Petri:

- Käyttöliittymän suunnittelu
- Pakettiautomaattien xml-tiedoston käsittely
- Lokitiedoston käsittely
- Virheentarkistus

Ohjelmaa koodattiin ketterän kehityksen mukaisesti, joten koodasimme välillä ristiin ja joissain ominaisuuksissa on molempien käden jälkeä.

Arviointi

Mielestämme ohjelmamme ansaitsee 40 pistettä. Siitä löytyy kaikki perusvaatimukset, kaksi laajennettua ja kaksi laajimman vaatimustason ominaisuutta. Ohjelma on koodattu asiallisesti ja se noudattaa asiallista tyyliä.

LIITTEET

Täydellinen luokkakaavio

Täydellinen luokkakaavio

katsottavissa selvempänä <http://ilee.gq/images/diagram.svg>

