OOP rühmatöö kirjeldus

1. Autorite nimed;

Jaanis Soosaar, Anastasia Starintšikova

1. Projekti põhjalik kirjeldus, kus on kirjas programmi eesmärk ja selgitus programmi Üldisest tööst, vajadusel lühike kasutusjuhis;

Meie projekti eesmärk on luua sõnade arvamise mäng. Programm valib meie loodud sõnad.txt failist suvalise sõna ning mängija hakkab seda ära arvama. Mängija peab pakkuma erinevaid tähti, mis võivad sõnas esineda. Kui mängija pakub sõnas esineva tähe, siis muutub sõna arvamine nö. kergemaks, tulevad välja pakutud tähed ning nende asukohad. Lisaks tuleb “poomispuule“ üks osa juurde. Mängija võidab, kui ta on ära arvanud kõik sõnas esinevad tähed (arvas sõna ära). Mängija kaotab, kui ta pakub 10 korda vale tähte ja “poomispuu“ saab lõplikuks.

1. Iga klassi kohta eraldi selle eesmärk ja olulisemad meetodid;

Kokku on 2 klassi: graafilineMäng ja graafika.

GraafilineMäng klassis on 6 meetodit: main, start, mänguRestart, taust1, getWord ja alustaMänguga.

* main – käivitab mängu.
* start – selles meetodis paneme paika peaaegu kogu mängu struktuuri.
* mänguRestart – meetod, mis käivitatakse mängu alustamise nupu vajutamisel ja see taaskäivitab mängu kõik elemendid.
* taust1 – paneb avalehele paika taustapildi.
* getWord – meetod, mis genereerib juhuslikult ühe sõna sõnad.txt failist.
* alustaMänguga – see meetod kontrollib mängija sisestust ja kas pakutud täht esineb arvatavas sõnas. Lisaks teeb muud sellega seotud graafilised toimingud.

Graafika klassis on 3 meetodit: joonistaPoomine, arvamiseJooned, paigutaTekst. Need kõik 3 keskenduvad mängu graafilisele poolele.

* joonistaPoomine – sätib paika poomispuu.
* arvamiseJooned – sätib paika arvamise jooned, kus hakatakse pooleli olevat sõna näitama.
* paigutaTekst – sätib paika tähed, mis asuvad arvamise joonte peal.

1. Projekti tegemise protsessi kirjeldus (erinevad etapid ja rühmaliikmete osalemine neis);

Suurema osa tööst tegime iseseisvalt kodudes, kuid kord nädalas saime kokku ning arutasime tekkinud probleemide ja küsimuste üle ning proovisime need lahedada.

1. Iga rühmaliikme panus (sh tehtud klassid/meetodid) ja ajakulu (orienteeruvalt);

Alguses kirjutasime kõik koodi start meetodisse ning hiljem jaotasime koodi laiali erinevatesse meetoditesse. Üldiselt oli esialgselt tööjaotus selline, et Anastasia tegeles avalehe stseeniga ning Jaanis alustas mängu stseeniga. Kuna pidi palju asju uurima googlest ning sageli kasutama katse-eksitus meetodit, siis arvame, et selleks hetkeks võis ajakulu olla kuni 7/8h. Kui tähtsam osa koodist oli start meetodis valmis, siis hakkasime seda nö sorteerima ja erinevatesse meetoditesse jagama, et kõik oleks arusaadavam ning et oleks kergem koodi debuggida ning testida. Tegime start meetodist klassi graafika meetodid ning joonistaPoomine ja alustaMänguga meetodid. Meetodid taust1 ja getWord meetodid on täielikult Anastasia tehtud. Meetodi mänguRestart pani kokku ja tegi korda Jaanis. Selle kõige peale läks veel lisaks orienteeruvalt 8 tundi.

1. Tegemise mured (nt millistest teadmistest/oskustest tundsite projekti tegemisel puudust);

Anastasia – Üldiselt oli JavaFX päris arusaamatu ja raske oli leida vajalikke asju, mis teeksid täpselt seda, mida hetkel vaja oli.

Jaanis – Muresid tekkis jooksvalt palju, kuid peaaegu kõik olid lahendatavad kiirete googeldamistega. Eriti oli probleemiks stackpane ja borderpane, ühesõnaga elementide paigutamine soovitud kohtadele.

1. Hinnang oma töö lõpptulemusele (millega saite hästi hakkama ja mis vajab arendamist);

Nagu ka esimeses osas, arvame veelgi, et alati saab vaeva näha sõnad.txt failiga, et arvatavad sõnad oleks reaalsemad ning paremad. Muidu arvame, et saime hästi hakkama, tegime põhimõtteliselt selle, mida plaanisime. Kindlasti on veel palju ruumi programmi arendamiseks. Kuna ajaga läks veidi kitsaks, siis isegi praegu ei jõudnud kõiki oma mõtteid realiseerida. Aga üldiselt jäime programmiga rahule.

1. Selgitus ja/või näited, kuidas programmi osi eraldi ja programmi tervikuna testisite ehk kuidas veendusite, et programm töötab korrektselt.

Programmi testisime jooksvalt, kui midagi valmis saime, siis kohe testisime ning proovisime vead üles leida. Meie programm vajas meie arvates palju testimist, sest kasutame programmis mitut osa, mis peavad korraga uuenema ning nö koostööd tegema.