

Mobile Project

Bluetooth Valokuvakehys

Antti Tuovinen
Valtteri Seuranen

12 2017
Tieto- ja viestintätekniikka
Tekniikan ala

1 Johdanto

Tämä projekti on ”Mobile Project” -kurssin harjoitustyö, jonka tarkoituksena oli kehittää opiskelijoiden taitoja Android- ja mobiiliohjelmistojen kehittämisessä.

Projektin tuloksena kehitimme kaksi osuutta: Android-hallintaohjelma, joka Bluetooth-yhteyttä käyttäen pystyy siirtämään valokuvia ja hallitsemaan valokuvakehystä. Toisena osuutena kehitimme Python-ohjelmointikielellä valokuvakehysohjelmiston, joka toimisi ”Raspberry Pi”- tai kannettavalla-tietokoneella. Valokuvakehys vastaanottaa puhelimelta lähetettyjä kuvia, joita kehys alkaa automaattisesti näyttämään. Lisäksi puhelin lataamaan valokuvakehyksellä näytettävän kuvan puhelimeensa Bluetooth-yhteydellä.

Projekti onnistui kohtalaisen hyvin. Mitään erityisiä ongelmia ei projektin kehittämisen aikana ilmentynyt ja aikaa vieviä bugeja ohjelmassa ei esiintynyt.

2 Tavoitteet

Kuten johdannossa kerrottiin, projektin tavoitteen oli kehittää opiskelijoiden mobiilisovellusten kehitys- ja koodaustaitoja ja sen lisäksi projektin toteutuksen tarkoituksena oli jatkokehittää ”Android Application Development” -kurssilla opittuja Android-kehitystaitoja, eteenkin Bluetoothin ja sitä ympäröivien teknologioiden osalta.

Projektin tuloksena luotiin puhelinapplikaatio, joka kommunikoi Bluetooth-yhteyttä käyttäen kannettavan tai Raspberry Pi tietokoneessa asennetun valokuvakehysohjelmiston kanssa. Puhelimen avulla voidaan hallita valokuvakehyksen tilaa.

Android-ohjelmaan kehitimme suunnitelmassa pohdittujen tavoitteiden pohjalta ominaisuuksia:

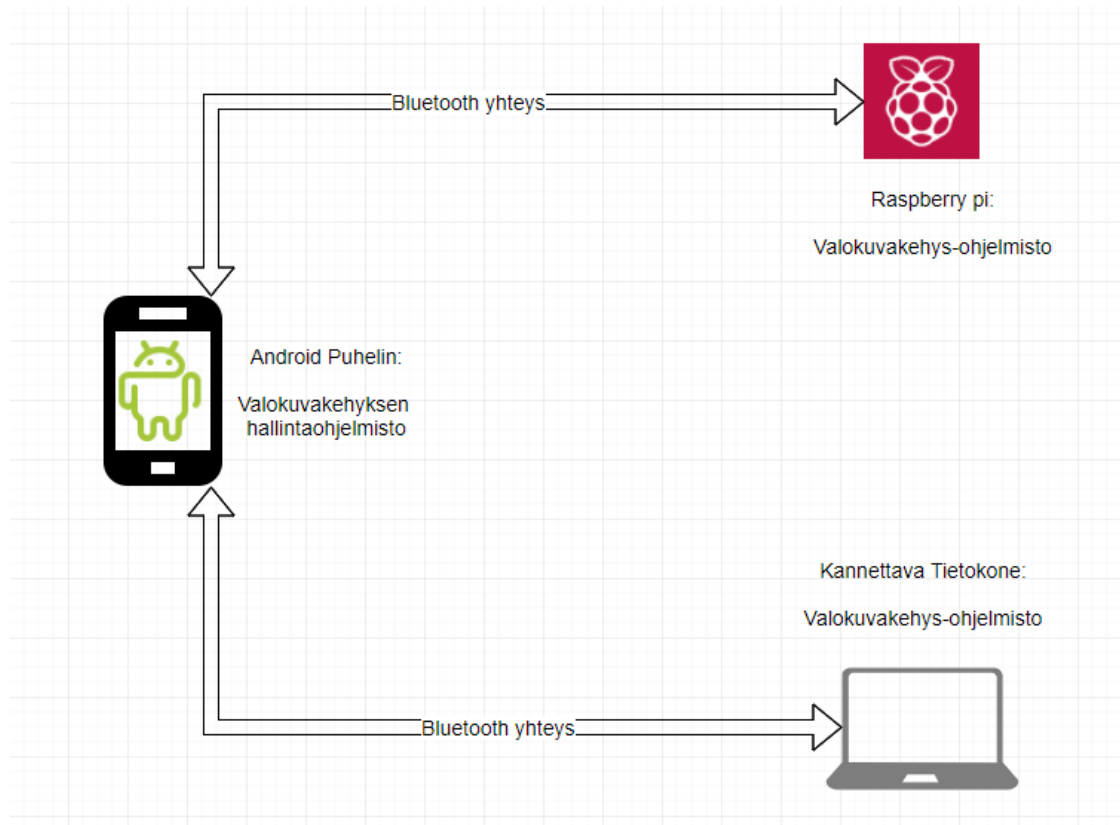
- Valokuvakehykseen siirrettävät kuvat voidaan hakea puhelimen galleriasta tai ottaa suoraan puhelimen kamerasta.
- Kuvien lataaminen valokuvakehykseen ja pois valokuvakehyksestä toimii.
- Android demoon tehtyjä Bluetooth-yhteyden hallinta toimintoja jatkokehitettiin paljon selkeämmiksi. Yhteyksien katkeamisen käsittely ja
- Lisättiin ohjelmaan ominaisuus puhelimen paljastamiseksi muiden laitteiden Bluetooth-laitteille.
- Kehykseltä ladatut valokuvat voidaan lisätä puhelimen galleriaan muistiin.

Valokuvakehyksessä olevan ohjelman on tarkoitus toteuttaa ainakin seuraavat ominaisuudet:

- Vastanottaa uuden kuvan Android-sovellukselta
- Lähettää nykyisen kuvan Android-sovellukselle, jossa se näytetään.
- Valokuvakehys pystyy sisältämään useita kuvia, joita voidaan painikkeiden avulla vaihdella.

3 Sovelluksen rakenne ja toteutus

Valokuvakehysohjelmisto koostuu 2 osasta: Android-puhelimesta, jossa on asennettuna valokuvakehyksen hallintasovellus, sekä kannettavasta- tai ”Raspberry Pi”-tietokoneesta, jossa on asennettuna valokuvakehyksen palvelinohjelmisto. Kuvassa 1 näkyy hahmoteltuna järjestelmässä käytetty laitteisto.



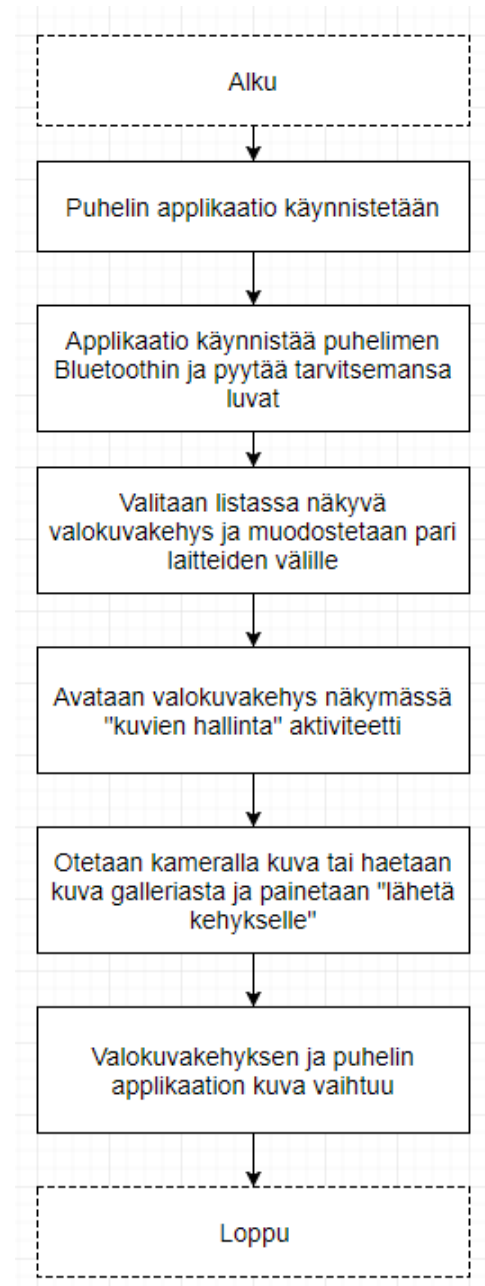
Kuva 1, Järjestelmässä käytettävä laitteisto

3.1 Käyttötapaus

Käyttötapauksena esimerkkinä: on olemassa "Raspberry Pi"-tietokone, joka on kiinnitettynä seinälle ripustettuun näyttöön. Tietokoneessa pyörii valokuvakehyksen palvelinohjelmisto ja tietokoneen Bluetooth on päällä.

Käyttäjä pystyy puhelinapplikaationsa avulla ottamaan Bluetooth-yhteydellä yhteyden seinällä olevaan valokuvakehykseen ja lähettämään ottamansa kuvan.

Kun kuva saapuu valokuvakehykseen valokuvakehys tallentaa saapuneen kuvan alkaa näyttämään sitä. Samalla aikaa puhelimen omassa näkymässä lähetetty kuva näytetään.

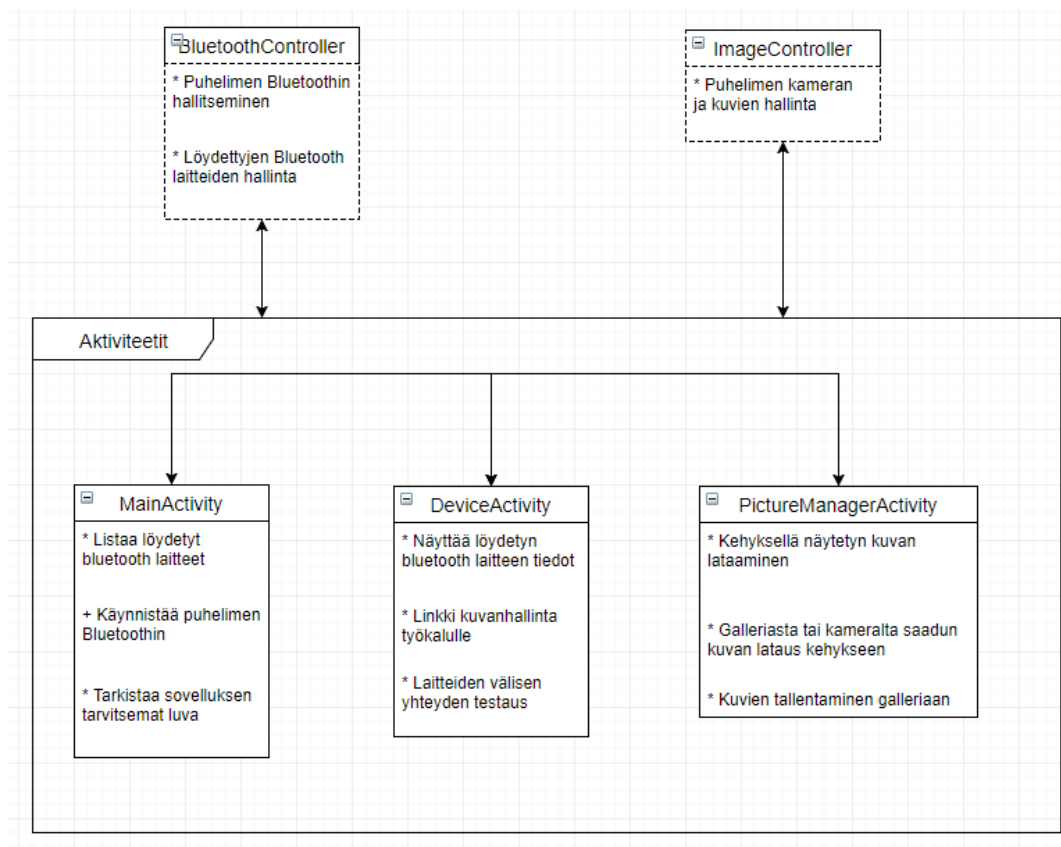


Kuva 2, Ohjelman käyttötapaus

4 Android-hallintaohjelma

Android-sovelluksen pohjana käytettiin ”Android Application Development ”–kurssilla tehtyä demoa, jota jatkokehitettiin, selkeytettiin ja paranneltiin.

Kuvassa 3 on mallinnettuna valokuvakehyksen hallintasovelluksen rakennetta ja luokkia. ”MainActivity”, ”Device Activity” ja ”PictureManagerActivity” ovat ohjelman pääaktiviteetteja, joiden avulla käyttäjä pystyy käyttämään ohjelman toimintoja. ”BluetoothController” ja ”ImageController” ovat yleisiä luokkia, joiden avulla puhelimen Bluetooth-yhteyksiä ja kameraa voidaan hallita.



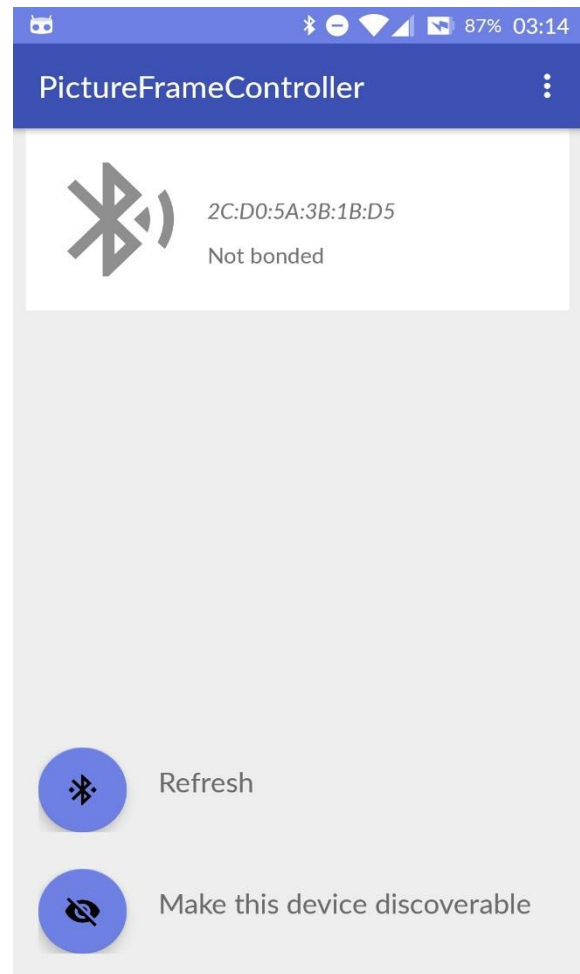
Kuva 3, Android hallintaohjelman rakenne

4.1 MainActivity

”MainActivity” on valokuvakehyksen hallintaohjelmiston aloitusnäky. Aktiviteetissä näytetään lista havaituista Bluetooth-laitteista ja onko puhelimen ja laitteen välillä pariyhteyttä (Bonding).

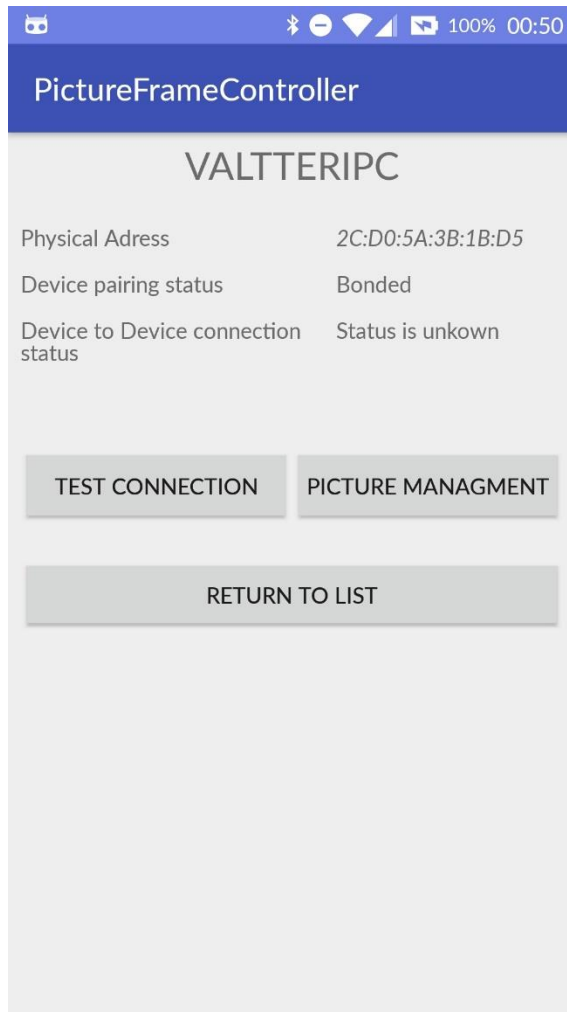
Pääaktiviteetissä myös tarkistetaan, että puhelimen Bluetooth on päällä ja ohjelman tarvitsemat luvat ollaan sille annettu. Bluetoothin hallintaan ja lupien tarkastamiseen ollaan käytetty ”BluetoothController”-luokkaa.

Kuva 4 on kuvankaappaus ”MainActivity”-aktiviteetista, kun puhelimelle ennestään tuntematon Bluetooth-laite ollaan löydetty.



Kuva 4, MainActivity näkymä

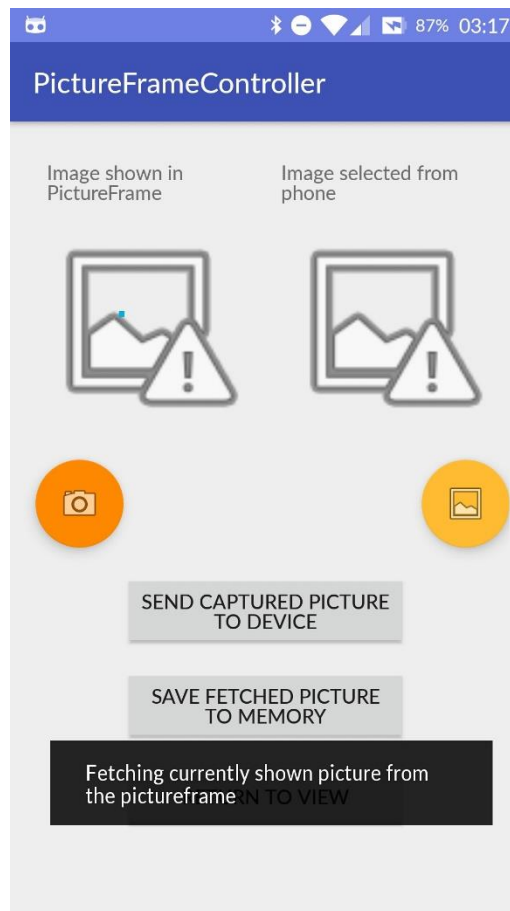
4.2 DeviceActivity



Kuva 5, DeviceActivity näkymä

"DeviceActivity"-näkymä kuvaa Bluetooth-laitteelta saatuja tietoja sekä mahdollistaa yhteyden muodostamisen testaamisen. Laitenäkymästä päästään myös kuvienhallinta osuuteen. Kuvassa 5 on otettu näyttökaappaus kannettavasta tietokoneesta, jossa on ollut valokuvakehyspalvelin ohjelmisto käynnissä.

4.3 PictureManagerActivity



Kuva 6, Kuvanhallintanäkymä

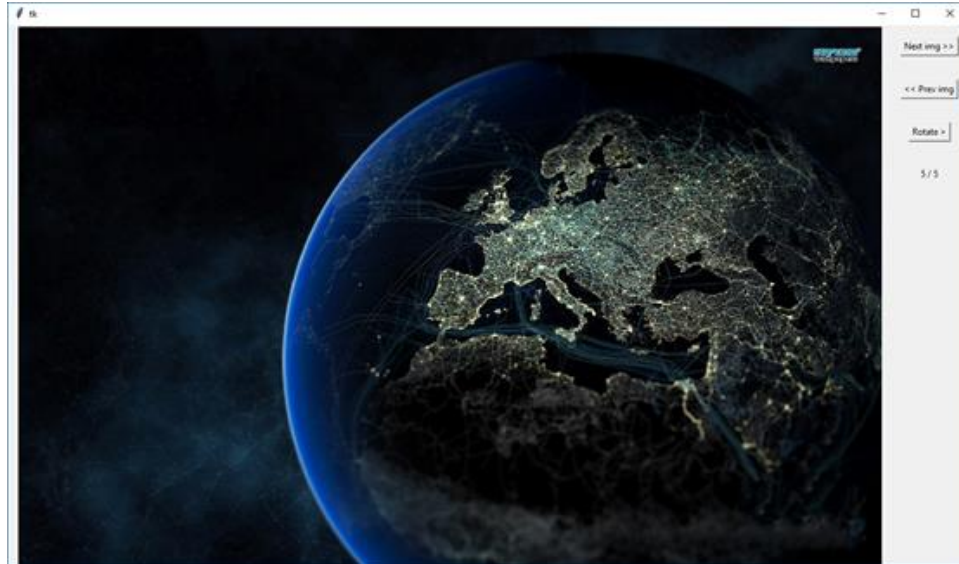
”Picture Manager Activity”-aktiviteetistä pystytään hallitsemaan valokuvakehyksessä näytettyjä valokuvia. Kuvassa 6 näkyvässä vasemmanpuolisessa kuvassa näytetään valokuvakehyksessä tällä hetkellä näkyvä kuva ja oikeanpuolisessa kuvassa näytetään valokuva, jonka käyttäjä on valinnut galleriasta tai ottanut puhelimen kameralla.

Kuvien alla olevista painikkeista voidaan käynnistää kamera- tai galleria-aktiviteetti kuvan valitsemista varten.

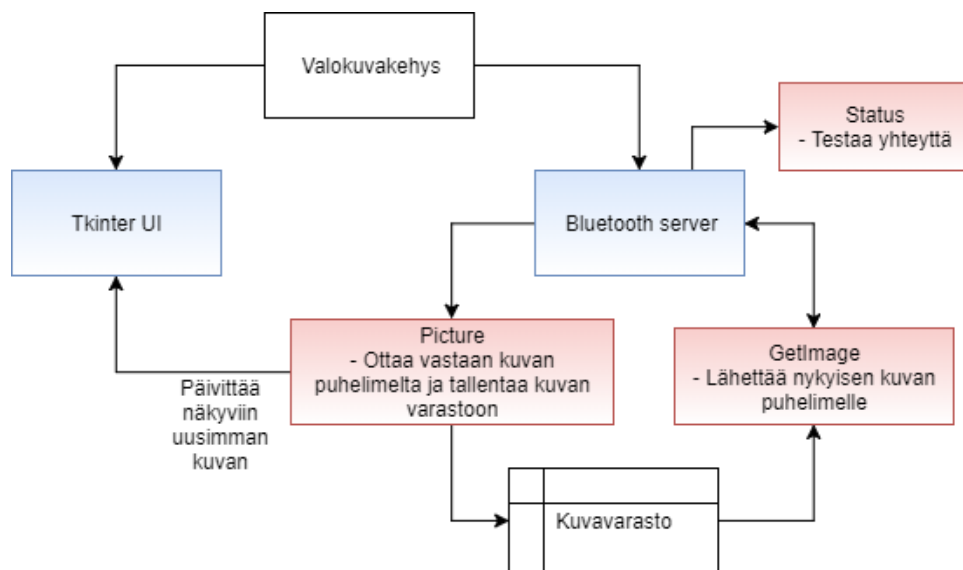
5 Valokuvakehys

Valokuvakehys tehtiin Pythonilla, jonka käyttöliittymä toteutettiin tkinter-kirjastolla. Kehyksen käyttöliittymä ja puhelimen kanssa kommunikoiva Bluetooth-backend päädyttiin laittamaan omiin threadeihin, jolloin niiden käyttäminen ei ole toisistaan riippuvaista, ja näin ohjelman käytettävyys parani.

Kehys-ohjelman toteuttaminen sujui hyvin. Suurimmaksi ongelmaksi muodostui kuvien lähettäminen puhelimelle muodossa, jota on helppo käsitellä javalla androidin puolella. Tämä ongelma päädyttiin ratkaisemaan lähettämällä kuva puhelimelle base64-enkoodattuna, jonka käsittelyyn löytyy androidista valmiiksi kirjasto.



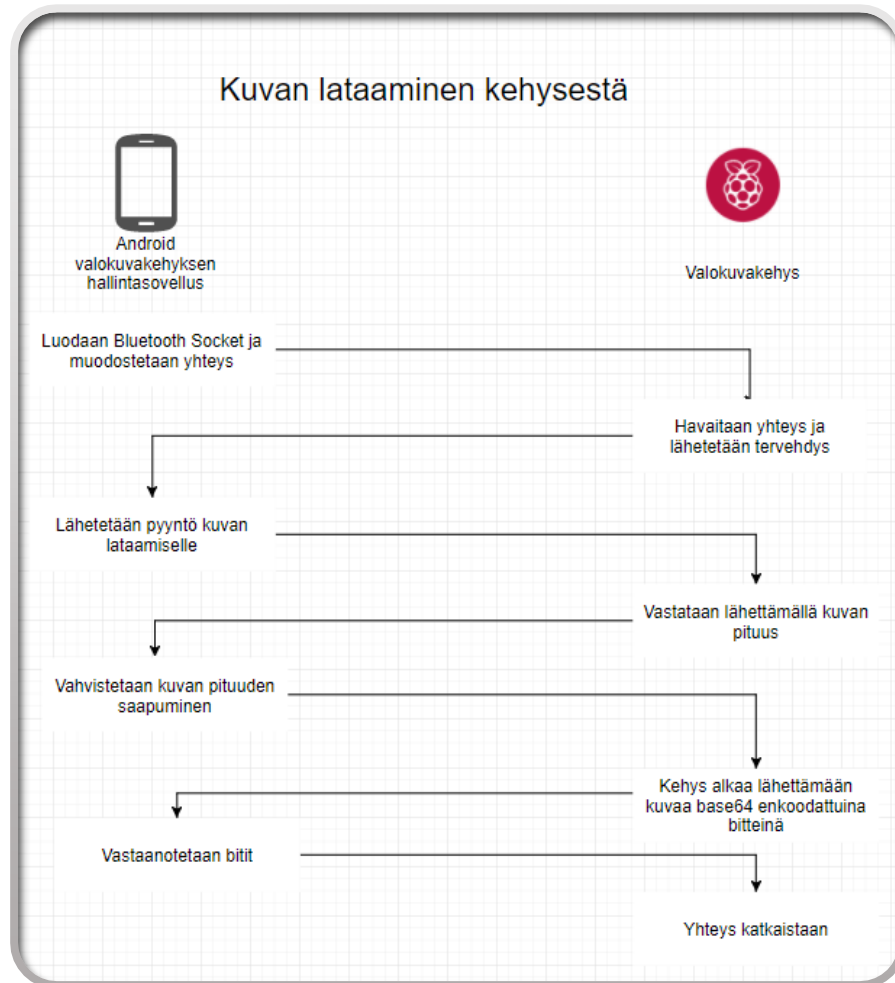
Kuva 7, Valokuvakehys



Kuva 5, Valokuvakehyksen rakenne

Kuvassa 8 on mallinnettuna Android-puhelimen ja valokuvakehyksen välinen kommunikaatio, kun Android haluaa ladata valokuvakehyksen näyttämän kuvan. Valokuvankehyksessä kuvien bitit siirretään laitteiden välillä, koodaamalla kuva base64-enkoodauksella.

Suurimpana ongelmana tiedonsiirrossa oli mahdolliset Bluetooth-yhteyden ongelmat esim. mitä tehdään, jos yhteys katkeaa laitteiden välillä kesken tiedonsiirron. Lisäksi ongelmana oli määrittää järjestelmä, jonka avulla laitteet tietäisivät, että tieto on päässyt perille.



Kuva 8, Hallintasovellus - Kehys, Kuvan lataaminen kehyksestä

6 Projektin työn kulku

Projektityön suunnitelmassa projektin työn kulku oltiin määritelty seuraavasti:

Taulukko 1, Työnkulun suunnitelma

Viikko	Tavoitteet
44	Projektin tekemisen aloitus. Projektisuunnitelma tehtynä.
45	Kuvan siirtäminen Android-laitteen ja valokuvakehyksen välillä toimii.
46	Kameralla kuvien ottaminen ja tallentaminen toimivat. Android ohjelman käyttöliittymä toimii.
47	Python käyttöliittymä valmis
48	Projektin esitelmän tekemisen aloitus.
49	Testausta ja projektiesitelmän tekoa
50	Projektiseminaari

Projektin aikana kirjattiin tarkemmin projektin kulkua wikisivuille, jossa työnkulku on kerrottuna osoitteessa:

https://github.com/Gitvalt/Bluetooth_Mobile_Project/wiki/Projektin_tyonkulku

Projektin aikatauluun muutoksia tuli viikkojen 48 ja 49 aikana, kun päätimme keskittyä enemmän muihin sen aikaisiin projekteihin ja kokeisiin.

7 Itsearviointit

7.1 Antti Tuovinen

Projekti laajensi omaa osaamista ja käsitystä varsinkin "alemman tason" Android-ohjelmoinnista, kun projektissa tutkittiin binääritiedostojen siirtämistä langattomasti.

7.2 Valtteri Seuranen

Omasta mielestäni tekemämme työ laajensi kattavasti Android-kurssin tutkimustyötä. Ohjelma täytti meidän sille laittamamme vaatimukset ja ohjelma kehittyi projektisuunnitelman mukaisesti.

Ohjelmassa olisi silti mahdollista kehittää vielä eteenpäin. Testauksen osalta ohjelma toimii, mutta tällä hetkellä uskon ohjelmassa olevan ongelmia ja mahdollisia tilanteita joita emme olleet käsitelleet riittävästi.