

Opinnäytetyö (AMK)

Tieto- ja viestintäteknikka

2026

Samu Nordman

Avoimen lähdekoodin ratkaisun hyödyntäminen fysiologisissa mittauksissa ja katseenseurannassa

– iMotions-pohjaisen ja avoimen ratkaisun vertailu



Opinnäytetyö (AMK) | Tiivistelmä

Turun ammattikorkeakoulu

Tieto- ja viestintäteknikka

2026 | 58 sivua

Samu Nordman

Avoimen lähdekoodin ratkaisun hyödyntäminen fysiologisissa mittauksissa ja katseenseurannassa

- iMotions-pohjaisen ja avoimen ratkaisun vertailu

Tässä opinnäytetyössä tarkasteltiin vaihtoehtoisia tapoja toteuttaa fysiologiset mittaukset ja katseenseuranta Turun ammattikorkeakoulun INSIGHT Labissa kuluttajaluokan laitteita ja avoimen lähdekoodin ohjelmistoja hyödyntäen. Tavoitteena oli selvittää, voidaanko iMotions-pohjaista ratkaisua osittain korvata kustannustehokkaammalla ja joustavammalla vaihtoehdolla mittausdatan laadun ja käytettävyyden säilyessä riittävänä. Taustalla oli tarve vähentää rajallisten lisenssien ja laitteiden kuormitusta.

Työssä toteutettiin 2 mittauskokonaisuutta: iMotions-ympäristö ja avoimeen lähdekoodiin perustuva ratkaisu, jossa keskeisenä komponenttina toimi Lab Streaming Layer -väliohjelmisto. Fysiologista dataa kerättiin Polar H10- ja Shimmer3 GSR+ -laitteilla, ja katseenseurantaa verrattiin Tobii Pro Fusion -laitteen ja verkkokamerapohjaisen EyeTrax -ratkaisun välillä.

Tulosten perusteella avoimen lähdekoodin ratkaisu ei täysin korvaa iMotions-järjestelmää erityisesti katseenseurannan tarkkuuden osalta, mutta fysiologisten mittausten laatu oli vertailukelpoinen. Ratkaisun etuja ovat kustannustehokkuus, modulaarisuus ja skaalautuvuus, minkä vuoksi se soveltuu täydentämään INSIGHT Labin mittausympäristöä.

Asiasanat:

Fysiologiset mittaukset, Katseenseuranta, Avoin lähdekoodi, Lab Streaming Layer

Bachelor's Thesis | Abstract

Turku University of Applied Sciences

Information and Communications Technology

2026 | 58 pages

Samu Nordman

Implementation of an open-source solution for physiological measurements and eye tracking

- A comparison between an iMotions-based and an open-source system

This thesis examined an alternative approach to implementing physiological measurements and eye tracking at the INSIGHT Lab of Turku University of Applied Sciences using consumer-grade devices and open-source software. The objective was to determine whether the existing iMotions-based solution could be partially replaced with a more cost-effective and flexible alternative while maintaining sufficient data quality and usability. The motivation for the study was the need to reduce the strain caused by limited licenses and equipment availability.

Two measurement setups were implemented: the existing iMotions environment and an open-source solution utilizing Lab Streaming Layer as the core middleware component. Physiological data were collected using Polar H10 and Shimmer3 GSR+ devices. Eye tracking performance was compared between the Tobii Pro Fusion device and a webcam-based EyeTrax solution.

The results indicate that the open-source solution does not fully replace the iMotions system, particularly in terms of eye-tracking accuracy. However, the quality of physiological measurements was comparable. The advantages of the proposed solution include cost efficiency, modularity, and scalability, making it suitable as a complementary measurement environment in the INSIGHT Lab.

Keywords:

Physiological measurements, Eye tracking, Open-source, Lab Streaming Layer

Sisältö

1 Johdanto	8
2 Tutkimuksessa käytetyt laitteet	10
2.1 Tobii Pro Fusion -katseenseurantalaite	10
2.2 Polar H10 -sykesensori	11
2.3 Shimmer3 GSR+ -sensori	12
2.4 Lenovo Performance FHD -verkkokamera	13
3 Tutkimuksessa käytetyt ohjelmistot	15
3.1 iMotions-tutkimusalusta	15
3.2 Lab Streaming Layer -väliohjelmisto	15
3.3 LabRecorder-tallennusohjelmisto	16
3.4 PolarBand2Isl-datankeräysohjelma	16
3.5 Shimmer C# Api -ohjelmistopaketti	16
3.6 EyeTrax-katseenseurantakirjasto	19
3.7 Open Broadcaster Software Studio -tallennussovellus	19
3.8 Inquisit 7 Lab -ohjelmistoympäristö	25
4 iMotions toteutus	29
4.1 Testiympäristön luominen iMotions -ympäristössä	29
4.2 Laitteiden yhdistäminen iMotions-ympäristöön	34
4.2.1 Polar H10-laitteen yhdistäminen	34
4.2.2 Shimmer3 GSR+ -laitteen yhdistäminen	37
4.3 Testin suoritus ja tallentaminen	39
4.4 Tallennettujen tietojen käyttö iMotions-ympäristössä	40
5 Uuden ratkaisun toteutus	42
5.1 Uuden ratkaisun komponentit	42
5.1.1 Polar H10 -laitteen yhdistäminen LSL -datavirtaan	42
5.1.2 Shimmer3 GSR+-laitteen yhdistäminen LSL-datavirtaan	43
5.1.3 EyeTrax- virtuaalikameran käynnistäminen	44
5.1.4 OBS videotallenteen aloitus	44

5.1.5 Inquisit	45
5.1.6 Lab Recorder	45
5.2 Tallennettujen tiedostojen käyttö	47
5.2.1 OBS -videotallenne	47
5.2.2 Lab Recorder XDF -tiedosto	48
6 Tulosten vertailu	50
6.1 Vertailuasetelma ja arviointiperusteet	50
6.2 Mittausdatan laatu	50
6.3 Datan synkronointi ja jatkokäytettävyys	51
6.4 Kustannukset ja skaalautuvuus	51
7 Lopuksi	52
Lähteet	54

Liitteet

Liite 1. Opinnäytetyön lähdekoodit

Kuvat

Kuva 1. Tobii Pro Fusion (Tobii, ei pvm.)	10
Kuva 2. Polar H10 -anturi ja rintavyö	11
Kuva 3. Polar H10 -laitteen käyttö. (Polar, ei pvm.)	12
Kuva 4. Shimmer3GSR+-laite käytössä	13
Kuva 5. Lenovo Performance FHD -verkkokamera (<i>Lenovo Performance FHD Webcam</i> , ei pvm.)	14
Kuva 6. Näytön kaappaus -lähteen lisääminen	20
Kuva 7. Näytön valitseminen lähteeksi	21
Kuva 8. Videokaappauslaitteen lisääminen lähteeksi	22
Kuva 9. Videokaappauslaitteen valitseminen	22
Kuva 10. Väriavain efektisuodattimen valitseminen	23

Kuva 11. Väriavaimen asetukset	23
Kuva 12. Lähteiden järjestys	24
Kuva 13. Yleiset kuva asetukset	25
Kuva 14. Uuden tutkimuksen luonti iMotions-ohjelmassa	29
Kuva 15. iMotions tutkimustyyppin valinta	30
Kuva 16. iMotions tutkimuksessa käytettävän näytön tiedot	31
Kuva 17. iMotions tutkimuksessa käytettävien sensoreiden valinta	32
Kuva 18. iMotions tutkimuksen nimeäminen	32
Kuva 19. Virikkeen lisääminen iMotions-ympäristöön	33
Kuva 20. iMotions-ympäristön virikkeen asetukset	34
Kuva 21. Sensori asetukset iMotions-ohjelmassa	35
Kuva 22. Uuden laitteen lisääminen iMotions-ohjelmaan	36
Kuva 23. Lisättävän laitteen tyyppi ja nimi	36
Kuva 24. Polar H10 -laitteen asetukset	37
Kuva 25. Shimmer-laitteen asetukset	38
Kuva 26. Vastaajan lisääminen iMotions-tutkimukseen	39
Kuva 27. Vastaajan tiedot	39
Kuva 28. Tallenteen avaaminen iMotions-ohjelmassa	40
Kuva 29. iMotions-tallenteen näkymä	41
Kuva 30. Toimiva yhteys PolarH10-laitteeseen	43
Kuva 31. Shimmer C# Api komentokehote	44
Kuva 32. Inquisit odottaa LSL-tallentajaa	45
Kuva 33. Lab Recorder tallennustietojen syöttäminen	46
Kuva 34. Datavirrat havaittuna Lab Recorder-sovelluksessa	46
Kuva 35. OBS -videotallenne	47

Kuviot

Kuvio 1. Yhdistetty kuvio	48
Kuvio 2. Zoomattu kuvio	49

Käytetyt lyhenteet

EKG	Elektrokardiogrammi, sydänsähkökäyrä (<i>Lyhenneluettelo - Kielikello, 2009</i>)
GSR	Galvatic skin response – Galvaaninen ihoreaktio, mittaus ihon sähkönjohtavuuden muutoksista, jotka heijastavat autonomisen hermoston aktivaatiota. (Can ym., 2019)
LSL	Lab Streaming Layer - LSL on avoimen lähdekoodin väliohjelmisto, joka suoratoistaa, vastaanottaa, synkronoi ja tallentaa erilaisista anturilaitteista hankittuja datavirtoja. (Kothe ym., 2025)
OBS	Open Broadcaster Software - Avoimen lähdekoodin ohjelmisto videotallennukseen ja suoratoistoon. (<i>Open Broadcaster Software OBS, 2025</i>)

1 Johdanto

Fysiologisia mittauksia ja katseenseurantaa hyödynnetään laajasti käyttäjäkokemuksen ja terveysteknologian tutkimuksessa (Ball & Richardson, 2023). Fysiologisilla mittauksilla, kuten sykkeen ja ihon sähkönjohtavuuden mittauksella, voidaan tarkastella autonomisen hermoston reaktioita erilaisiin virikkeisiin (Ball & Richardson, 2023). Katseenseurannan avulla puolestaan analysoidaan, mihin kohteeseen henkilön katse kohdistuu milläkin hetkellä (RealEye, ei pvm.). Katseenseurantaa voidaan toteuttaa useilla eri menetelmillä, kuten päähän kiinnitettävillä laitteilla, silmälaseihin integroiduilla ratkaisuilla tai etäseurantana näytön yhteyteen sijoitetun kameran avulla (RealEye, ei pvm.).

Turun ammattikorkeakoulun INSIGHT Lab toimii oppimis- ja tutkimusympäristönä, jossa opiskelijat ja tutkimusryhmät toteuttavat mittauksia kurssitehtävissä, opinnäytetöissä ja tutkimushankkeissa (Turku AMK, 2026). Laboratoriossa hyödynnetään iMotions -ohjelmistoa sekä Tobii Pro Fusion -katseenseurantajärjestelmää (Turku AMK, 2025). Ratkaisu perustuu rajallisiin lisensseihin ja yhteen työasemaan, mikä aiheuttaa kuormitusta erityisesti opetuskäytössä.

Tämän opinnäytetyön tavoitteena on selvittää, voidaanko iMotions -pohjaista mittausratkaisua osittain korvata kustannustehokkaammalla ja joustavammalla avoimeen lähdekoodiin perustuvalla vaihtoehdolla. Tarkoituksena on arvioida, onko uuden ratkaisun mittausdatan laatu ja käytettävyys käyttötarkoitukseen riittäviä. Keskeisenä tekijänä ratkaisussa hyödynnetään Lab Streaming Layer (LSL) -väliohjelmistoa, joka mahdollistaa eri laitteiden tuottamien datavirtojen synkronoidun keräämisen ja tallentamisen (Kothe ym., 2025). LSL on valittu keskeiseksi tekijäksi, koska se on ilmainen ja monet erilaiset laitteet ja ohjelmistot tukevat sitä.

Työn teoreettinen viitekehys perustuu katseenseurantaa ja fysiologisia mittauksia käsittelevään kirjallisuuteen, ja teknisessä toteutuksessa hyödynnetään avoimen lähdekoodin ohjelmistoja ja niiden dokumentaatioita.

Työssä toteutetaan vertailu nykyisen iMotions -pohjaisen ratkaisun ja avoimen lähdekoodin mittauskokonaisuuden välillä. Vertailussa tarkastellaan katseenseurannan toimivuutta, fysiologisten mittausten laatua, järjestelmän käyttäjäystävällisyyttä sekä kustannus- ja skaalautuvuustekijöitä. Tässä työssä keskitytään etäkatseenseurantaan, jossa käyttäjään ei kiinnitetä erillistä laitteistoa katseenseurannan osalta. Työssä ei tarkastella kliinisen tason validointia. Kaikki testit on toteutettu Windows 11 -ympäristössä.

Ennen työn aloittamista hypoteesi on, että vaihtoehtoinen ratkaisu on toteutettavissa, mutta sen tulosten mittausdatan laatu ja käyttäjäkokemus on epävarmaa. Työ etenee siten, että ensin esitellään tutkimuksessa käytetyt laitteet ja ohjelmistot. Tämän jälkeen esitellään nykyinen iMotions-pohjainen toteutus ja uusi ratkaisu. Lopuksi verrataan molempien ratkaisujen toteutusta ja niiden tuloksia.

2 Tutkimuksessa käytetyt laitteet

Suuri osa käytetyistä laitteista ovat samoja, joita käytetään jo olemassa olevassa työasemassa (INSIGHT Lab, 2025). Kaikki laitteet ovat jo käytettävissä INSIGHT Labin tiloissa. Kaikilla laitteilla mitataan fyysisiä ominaisuuksia ihmiskehosta ja seurataan reagoimista virikkeisiin. Kyseiset laitteet on valittu, koska ne tukevat avoimenlähdekoodin LSL-ratkaisuja ja ovat käytössä aikaisemmissa tutkimus ja opetuskohteissa. Suurimmat erot uuteen ratkaisuun ja olemassa olevaan tilanteeseen liittyy pitkälti käytettäviin ohjelmistoihin. Näistä eroista kerrotaan lisää luvussa 3.

2.1 Tobii Pro Fusion -katseenseurantalaite

Tobii Pro Fusion on ammattilaiskäyttöön suunniteltu näyttöpohjainen katseenseurantalaite (Tobii, ei pvm.). Laitteessa on 2 silmiä seuraavaa kameraa ja 250 Hz:n näytteenottotaajuus, jonka avulla saadaan riittävän tarkkoja katseenseuranta tuloksia. (Kuva 1.) (Tobii, ei pvm.).



Kuva 1. Tobii Pro Fusion (Tobii, ei pvm.)

Laitetta käytetään tällä hetkellä useassa eri tutkimuksessa ja kurssitehtävien suorittamiseen INSIGHT Labissa (Turku AMK, 2026). Sitä on kuitenkin käytössä vain 1 kappale (INSIGHT Lab, 2025), mikä aiheuttaa sen käyttöön aikataulullisia haasteita, joita tässä työssä pyritään lieventämään. Jotta laitteen katseenseurannasta saadaan hyödyllisiä analyttisiä tuloksia, INSIGHT Labissa

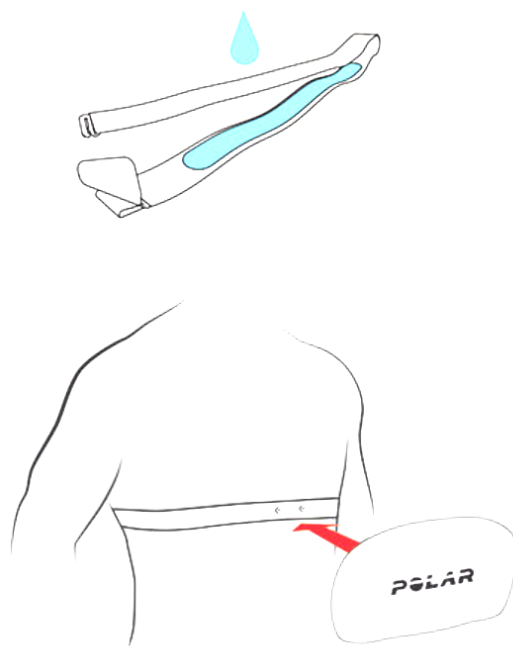
on käyttänyt siihen iMotions-ohjelmistoa (Turku AMK, 2025).
iMotions-ohjelmistosta kerrotaan tarkemmin luvussa 3.1.

2.2 Polar H10 -sykesensori

Polar H10 on kuluttajamarkkinoille suunniteltu sykkeen mittauslaite, joka käyttää elektrokardiogrammi (EKG) teknologiaa käyttäjän sydämen sykkeen mittaamiseen (Polar, 2025a). Laite kiinnitetään käyttäjän rintakehälle rintavyöllä, joka kuuluu laitteeseen (Kuva 2.) ja rintavyön sisäpinnalla oleva elektrodialue kostutetaan vedellä EKG -tarkkuuden saamiseksi (Kuva 3.) (Polar, 2025a).



Kuva 2. Polar H10 -anturi ja rintavyö



Kuva 3. Polar H10 -laitteen käyttö. (Polar, ei pvm.)

Laite yhdistetään tietokoneeseen Bluetooth-yhteydellä. Ohjelmistosta, jolla kerätään laitteesta tietoa, kerrotaan tarkemmin luvussa 3.4. Tämä laite valittiin tähän tutkimukseen, koska se on helppokäyttöinen (Polar, 2025b) ja INSIGHT Labissa on useampi kappale käytettävissä (INSIGHT Lab, 2025).

2.3 Shimmer3 GSR+ -sensori

Shimmer3 GSR+ -laite on ammattilaiskäyttöön suunniteltu puettava laite, jolla voidaan mitata useita metriikoita ja ihmiskehon ominaisuuksia (Pedersen,

2025). Laitteeseen kuuluu pääyksikön lisäksi 2 sormi-kuivaelektrodia johtoineen sekä rannehihna (Kuva 4.).



Kuva 4. Shimmer3GSR+-laite käytössä

Galvaaninen ihoreaktio (GSR) tarkoittaa ihon sähkönjohtavuuden muutosta, joka liittyy ihmisen vireystilaan ja autonomisen hermoston toimintaan (Can ym., 2019). Käytännössä tämä tarkoittaa ihon sähkönjohtavuuden muuttumista, joka johtuu hikirauhasten aktivoitumisesta (Can ym., 2019). Vaikka laitteella pystyisi mittaamaan useita metriikoita (Pedersen, 2025), tässä tutkimuksessa laitteesta kerättiin pelkästään GSR -arvoja. Myöskään arvojen syvällisempi analyysi ei kuulu tämän opinnäytetyön laajuuteen. Laite yhdistetään Bluetooth-yhteydellä tietokoneeseen. Ohjelmistosta, jolla kerätään laitteesta tietoa, kerrotaan tarkemmin luvussa 3.5.

Tämä laite valittiin tähän tutkimukseen, koska niitä on jo entuudestaan useampi kappale saatavilla INSIGH Labissa (INSIGHT Lab, 2025), ja niitä käytetään jo useassa muussa tutkimuksessa ja kurssitehtävissä (Turku AMK, 2025).

2.4 Lenovo Performance FHD -verkkokamera

Tässä tutkimuksessa yksi tärkeimmistä tutkimuskysymyksistä on, minkä laatuista tuloksia voidaan saavuttaa katseenseurannassa käyttäen

kuluttajamarkkinoille suunnuttua verkkokameraa, verrattuna ammattilaiskäyttöön suunniteltua katseenseuranta laitetta.

Tässä työssä on käytetty Lenovo Performance FHD -verkkokameraa, joka on 1080p- resoluution kamera (Kuva 5.) (*Lenovo Performance FHD Webcam*, ei pvm.), mikä on yleinen kuvalaatu kuluttajakäyttöön (*Webcam Buying Guide | CDW, 2022*).



Kuva 5. Lenovo Performance FHD -verkkokamera (*Lenovo Performance FHD Webcam*, ei pvm.)

Tämä laite valittiin tähän tutkimukseen, koska kyseistä verkkokameramallia on jo käytössä useassa INSIGHT Labin työpisteellä (INSIGHT Lab, 2025).

Ohjelmistoista, joilla saavutetaan verkkokameraa hyödyntävä katseenseuranta mahdollisuus, kerrotaan tarkemmin lisää luvussa 3.6.

3 Tutkimuksessa käytetyt ohjelmistot

3.1 iMotions-tutkimusalusta

iMotions on lisensoitu tutkimusalusta, jota käytetään ihmisen käyttäytymisen analysointiin biosensoreiden avulla (iMotions, ei pvm.). Se on ohjelmisto, joka synkronoi ja analysoi useita eri tietolähteitä samanaikaisesti (iMotions, ei pvm.-c). Sen avulla voidaan tutkia, miten ihminen reagoi fyysisesti erilaisiin virikkeisiin (iMotions, ei pvm.-c).

iMotions on tällä hetkellä käytössä INSIGHT Labissa useassa eri käyttötarkoituksessa (Turku AMK, 2025), mutta käytössä olevalla lisenssillä iMotions-ohjelma voi olla käytössä vain yhdessä tietokoneessa kerrallaan (iMotions, 2023). Tämän työn yksi keskeinen tutkimuskysymys on, kuinka tätä tutkimusalustaa voitaisiin korvata edullisemmalla vaihtoehdolla.

iMotions ympäristö on tämän opinnäytetyön suurin vertailukohde, johon uuden ratkaisun tuloksia verrataan. Tulosten vertailusta kerrotaan tarkemmin luvussa 6.

3.2 Lab Streaming Layer -väliohjelmisto

Lab Streaming Layer (LSL) on avoimen lähdekoodin väliohjelmisto, joka suoratoistaa, vastaanottaa, synkronoi ja tallentaa erilaisista laitteiden anturilaitteista ja ohjelmistoista hankittuja datavirtoja (Kothe ym., 2025). Sitä käytetään laajalti neurotieteessä ja psykologisissa tutkimuksissa, joissa kerätään tietoa erilaisista antureista ja ohjelmistoista samaan aikaan (Kothe ym., 2025). LSL avulla pystytään synkronoimaan kerättyjä datavirtoja millisekunnin tarkkuudella useasta eri lähteestä ja se tukee monia eri valmistajien laitteita ja ohjelmistoja. (Kothe ym., 2025).

Tässä työssä kerätään EKG, GSR ja katseenseuranta dataa. LSL:n yhteydessä käytetään usein LabRecorder-ohjelmaa kerättyjen tietojen tallentamiseen (Kothe ym., 2025).

3.3 LabRecorder-tallennusohjelmisto

LabRecorder on avoimen lähdekoodin sovellus, jolla voidaan tallentaa LSL-datavirrat yhteen XDF-tiedostoon myöhempää analyysiä varten (LabStreamingLayer, 2025a). XDF-tiedosto on yleiskäyttöinen tiedostomuoto suunniteltu LSL:lle (LabStreamingLayer, 2025a). XDF-tiedostoja voidaan käsitellä erilaisissa ympäristöissä, kuten Python, MATLAB, EEGLAB, BCILAB ja MoBILAB (LabStreamingLayer, 2025a). XDF-tiedostojen käsittelystä tässä työssä kerrotaan tarkemmin luvussa 5.2.2. LabRecorder on kokonaan osana uutta ratkaisua, eikä se ole käytössä nykyisessä toteutuksessa.

3.4 PolarBand2Isl-datankeräysohjelma

PolarBand2Isl on avoimen lähdekoodin sovellus, jolla voidaan lähettää Polar H10-laitteesta kerättyä EKG -dataa LSL -datavirtaan (Span, 2023). Sovellus etsii kaikki tietokoneeseen yhdistetyt Polar H10-laitteet Bluetooth-yhteydellä ja käyttäjä voi valita listasta haluamansa laitteen. Kun haluttu laite on valittu, sovellus alkaa lähettämään LSL -datavirtaa (Span, 2023), joka voidaan tallentaa LabRecorder-sovelluksen avulla (LabStreamingLayer, 2025a).

Tätä sovellusta käytetään uudessa ratkaisussa, eikä se ole aiemmin käytössä INSIGHT Labissa. Sovellus on ladattu suoraan sen Github-arkistosta, eikä siihen ole tehty muutoksia.

3.5 Shimmer C# Api -ohjelmistopaketti

Shimmer C# Api on avoimen lähdekoodin ohjelmistokehityspaketti, jota voidaan käyttää Shimmer3-laitteiden hallintaan ja datan lähettämiseen. (Shimmer

Engineering, 2025). Projektin virallinen GitHub-arkisto sisältää valmiita C#-ohjelmointikielellä tehtyjä esimerkkejä Shimmer3-laitteen käyttämiseen (Shimmer Engineering, 2025). Tässä työssä hyödynnetään projektin ShimmerPPGHRGSRConsoleAppExample-komnetokehotesovellusta ja sen valmiiseen ohjelmistokoodiin on tehty muutamia lisäyksiä, jotta halutut kerättävät tiedot saadaan lähetettyä LSL -datavirtaan.

Valmis esimerkkisovellus tulostaa komentokehoteeseen yhdistetyn Shimmer3-laitteen mittaamia aikaleimattuja 3 suuntaisia kiihtyvyyssanturiarvoja, fotoplethysmografi-arvoja (engl. photoplethysmography, PPG) ja galvaattista ihoreaktiota (GSR) (Shimmer Engineering, 2025). Tässä työssä kerätään kuitenkin vain GRS-arvoja.

Tätä ohjelmistoa käytetään uudessa ratkaisussa, eikä se ole aiemmin käytössä INSIGHT Labissa. Se on valittu käytettäväksi, koska sen avoimeen lähdekoodiin voidaan tarvittaessa itse tehdä muutoksia. GSR -mittausten tallentaminen edellyttää vielä, että ohjelmistokoodiin tehdään tarvittavat lisäykset LSL -datavirran luomiseksi.

Projektiympäristöön täytyy lisätä tarvittavat NuGet-paketit:

- SharpLSL
- SharpLSL.Native.win-x64.

NuGet on .Net-alustan virallinen pakettienhallintajärjestelmä, jonka avulla sovelluskehittäjät voivat helposti jakaa uudelleenkäytettävää ohjelmistokoodia (Microsoft, 2022). Nämä NuGet-paketit tarjoavat tarvittavat toiminnot LSL -datavirran luomiseksi (LabStreamingLayer, 2025b). NuGet-pakettien lisäämisen jälkeen ohjelmistokoodiin lisätään vielä tarvittavat muutokset (Ohjelma 1.).

Ohjelma 1. ShimmerPPGHRGSRConsoleAppExample muokattu ohjelmakoodi

```

...
using SharpLSL;

...

public void start() // sisälle lisätään:
{
    // Initialize LSL stream for GSR
    StreamInfo info = new StreamInfo(
        "ShimmerGSR",
        "GSR",
        1,
        SamplingRate,
        SharpLSL.ChannelFormat.Double,
        "shimmer3gsr"
    );

    outlet = new StreamOutlet(info);

    ...

    // metodin sisälle lisätään:

    public void HandleEvent(object sender, EventArgs args)
    ...

    // --- LSL SEND GSR SAMPLE ---
    if (outlet != null)
    {
        double[] sample = new double[] { dataGSR.Data };
        outlet.PushSample(sample);
    }

    ...

```

Näillä lisäyksillä luodaan LSL-datavirta ja datavirtaan lähetetään halutut GSR-arvot.

Alkuperäinen ohjelmakoodi vaatii, että käytettävän Shimmer3-laitteen käyttämä COM-portti on kiinteästi koodattu ohjelmakoodiin laitteen ja ohjelman yhdistämiseen (Shimmer Engineering, 2025). Tätä ominaisuutta on muokattu tämän opinnäytetyön tarkoitukseen. Muokattu ohjelma käy läpi kaikki käyttöjärjestelmän käytössä olevat COM-portit ja etsii niistä Shimmer3-laitteen käyttämää porttia ja yhdistää kyseiseen porttiin automaattisesti. Käytetty ohjelmakoodi on kokonaisuudessaan liitteessä 1.

Kun tarvittavat lisäykset ja muutokset on tehty, tarvitsee ohjelma vielä rakentaa uudelleen koodieditorissa. Kun ohjelmistokoodi on rakennettu onnistuneesti,

voidaan se suorittaa, jolloin avautuu komentokehote ja ohjelma yhdistää automaattisesti Shimmer3-laitteeseen ja aloittaa tiedon keräämisen ja LSL -datavirran lähettämisen (Shimmer Engineering, 2025).

3.6 EyeTrax-katseenseurantakirjasto

EyeTrax on avoimen lähdekoodin Python-kirjasto, jonka avulla voidaan luoda reaaliaikaista katseenseurantaa käyttäen tavallista verkkokameraa. (Zhang, 2025). Kirjasto hyödyntää valmiiksi opetettuja konenäkömalleja tunnistamaan kasvat, silmät ja pupillit videokuvasta (Zhang, 2025). Kalibroinnin aikana opetetaan tekoälymalli automaattisesti yhdistämään tietyt kasvojen ja silmienasennot vastaamaan tiettyjä näytön koordinaatteja (Zhang, 2025). Tämä kirjasto tukee myös virtuaalikameraa (Zhang, 2025) jota hyödynnetään katseenseurannan tallentamiseen käyttäen Open Broadcaster Software Studio-ohjelmaa, josta kerrotaan tarkemmin luvussa 3.7. EyeTraxiä käytetään suoraan komentokehoteen kautta ja suoritettavan tiedoston yhteydessä voidaan käyttää valinnaisia kehoteasetuksia, kuten erilaisia katseenseurannan suodattimia ja kalibrointi tarkkuuden määritelmiä (Zhang, 2025).

Tämä Python-kirjasto on osana uutta ratkaisua, eikä sitä ole käytetty aiemmin INSIGHT Labissa. Ohjelmakoodi, jota käytetään tässä työssä, on ladattu suoraan kirjaston Github-arkistosta (Zhang, 2025), eikä siihen ole tehty muutoksia.

3.7 Open Broadcaster Software Studio -tallennussovellus

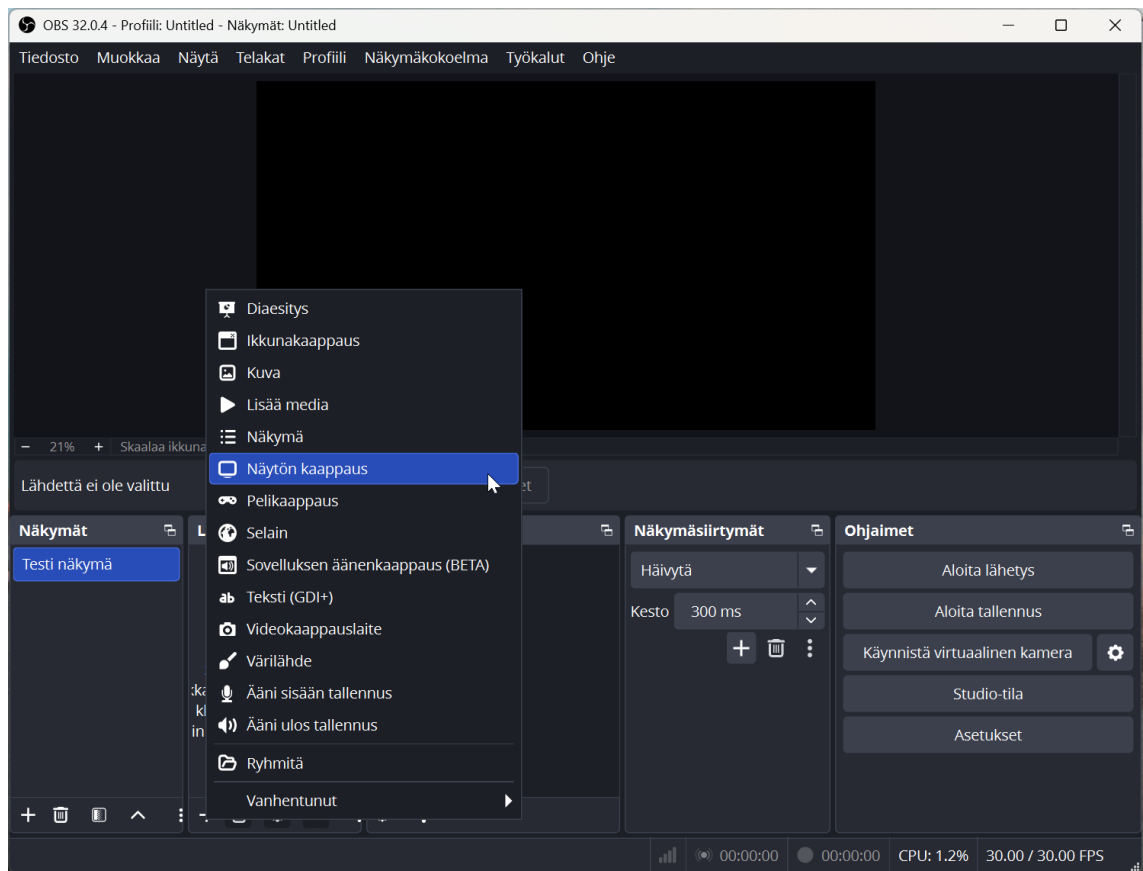
Open Broadcaster Software Studio (OBS) on avoimenlähdekoodin sovellus, jota voidaan käyttää suoratoistoon ja videotallennukseen. (*Open Broadcaster Software | OBS*, 2025). OBS-studiossa voidaan luoda näkymiä, joihin valitaan elementit, joita halutaan videotallentaa. (*Open Broadcaster Software | OBS*, 2025). Tässä työssä hyödynnetään OBS -ominaisuutta, jolla voidaan käyttää virtuaalikameraa lähteenä (*Open Broadcaster Software | OBS*, 2025) ja

lähteenä tässä työssä käytetään EyeTrax-kirjastolla luotua virtuaalikameraa (Zhang, 2025). Katseenseurannan tallentaminen näytölle edellyttää uuden näkymän luomista OBS Studioon sekä tarvittavien lähteiden liittämistä. Lisäksi EyeTrax-virtuaalikameralähteeseen on määritettävä oikeanlainen väriavainnussuodatin.

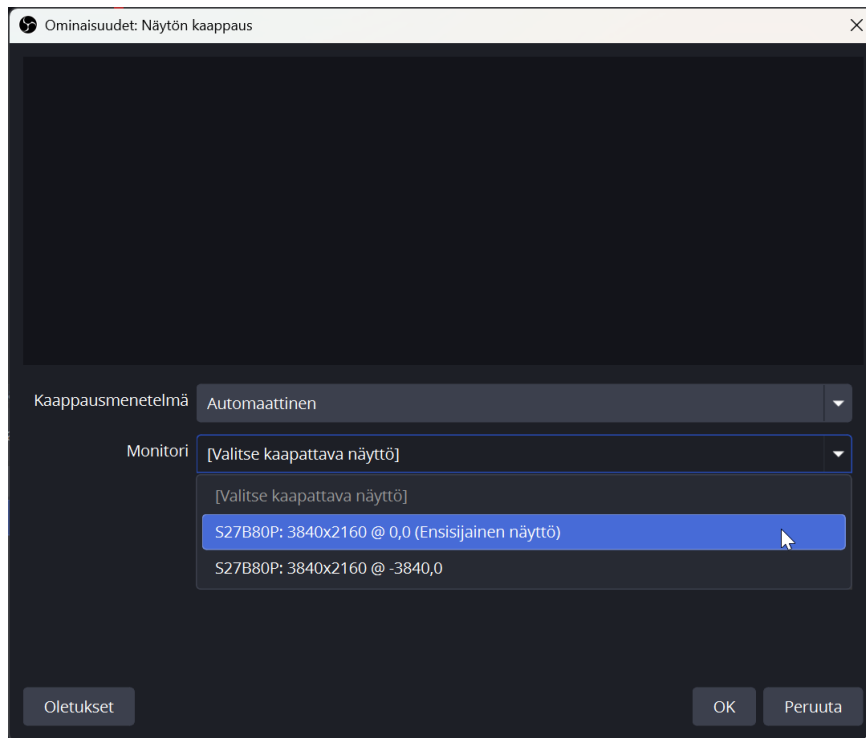
Tätä ohjelmistoa käytetään uudessa ratkaisussa testin videotallentamiseen, eikä se ole aiemmin käytössä INSIGHT Labissa. Se valittiin käytettäväksi, koska se on ilmainen ja se tukee virtuaalikameroita lähteenä.

Jotta katseenseurantaa voidaan videotallentaa onnistuneesti, tarvitsee OBS Studioon luoda oikeanlainen näkymä ja EyeTrax-virtuaalikamera pitää olla käynnistettynä.

Näkymän lähteeksi valitaan näytön kaappaus (Kuva 6.) ja valitaan haluttu tietokoneen näyttö (Kuva 7.), jos tietokoneeseen on kytketty useampi näyttö.

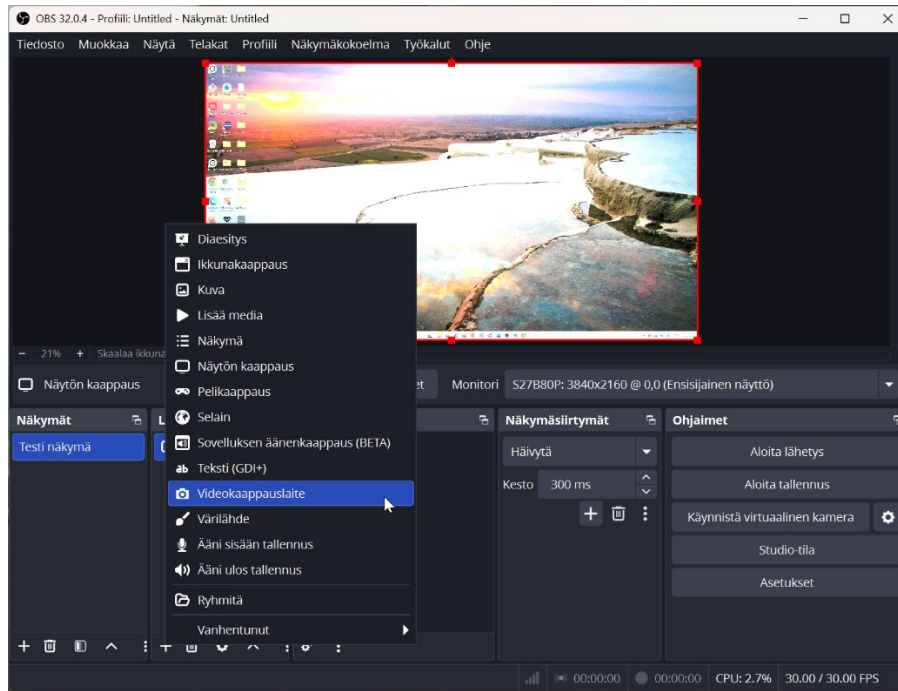


Kuva 6. Näytön kaappaus -lähteen lisääminen

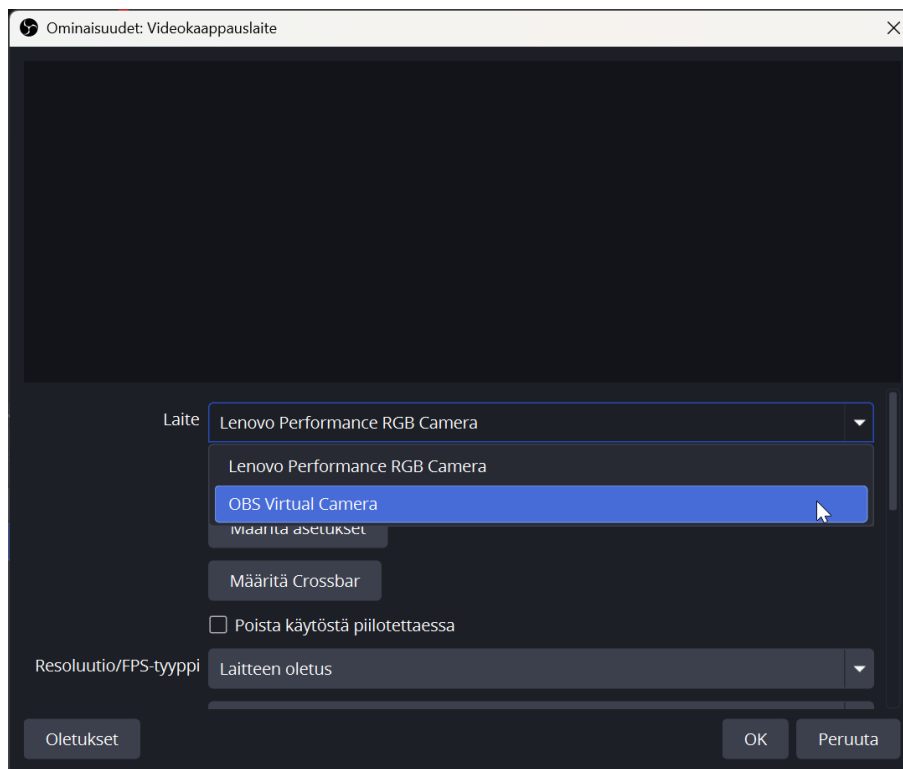


Kuva 7. Näytön valitseminen lähteeksi

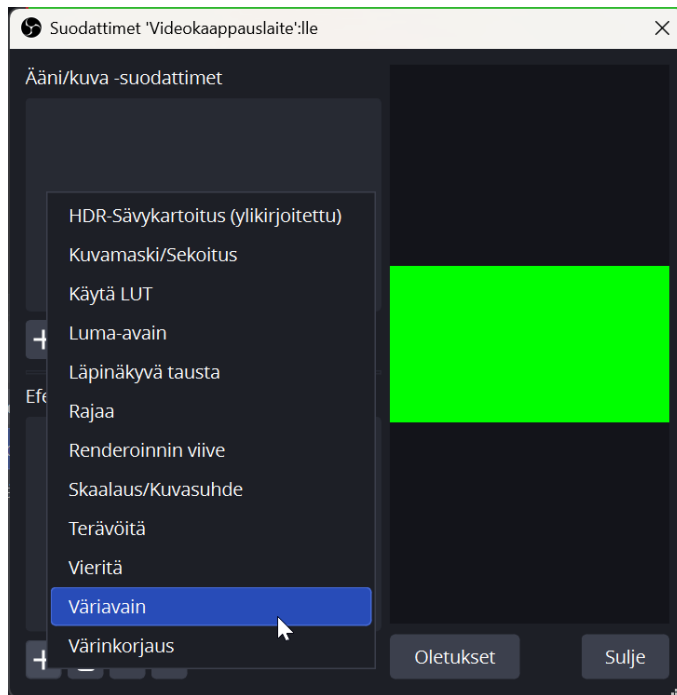
Seuraavaksi lisätään lähteeksi videokaappauslaite (Kuva 8.) ja valitaan laitelistasta OBS Virtual Camera (Kuva 9.) Tämän jälkeen videokaappauslaite-lähteeseen lisätään suodattimeksi efektisuodatinlistasta väriavain (Kuva 10.). Väriavaimen asetukset tulee vielä säätää sopiviksi, jotta virtuaalikameran vihreä tausta katoaa läpinäkyväksi ja oranssi katseenseurantakohdepiste jää näkyviin (ck-zhang, 2025). Tähän yleensä riittää, kun muuttaa samanlaisuusasetuksen arvoksi 160 (Kuva 11.).



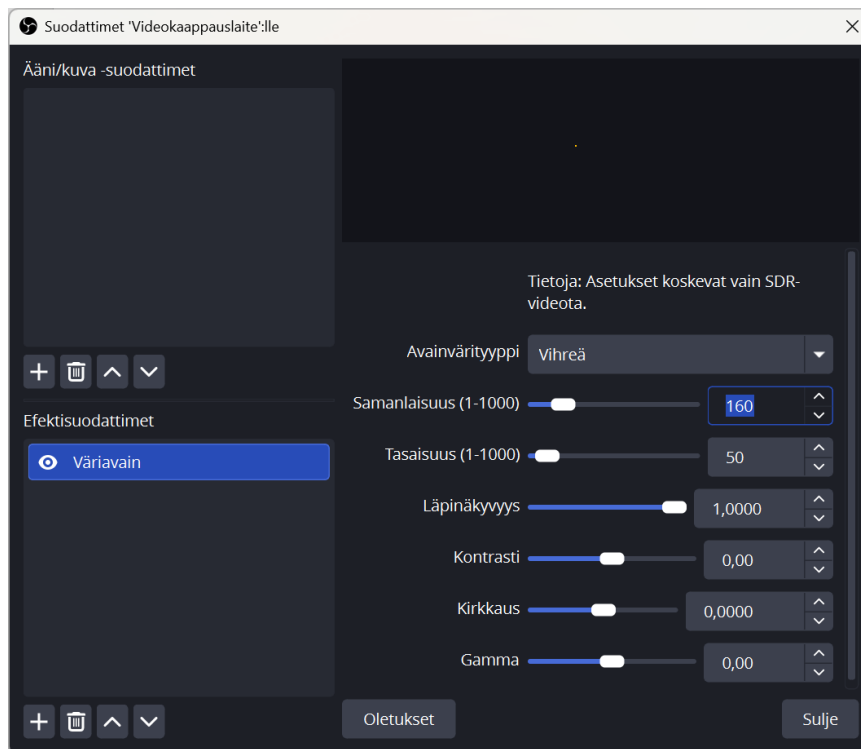
Kuva 8. Videokaappauslaitteen lisääminen lähteeksi



Kuva 9. Videokaappauslaitteen valitseminen

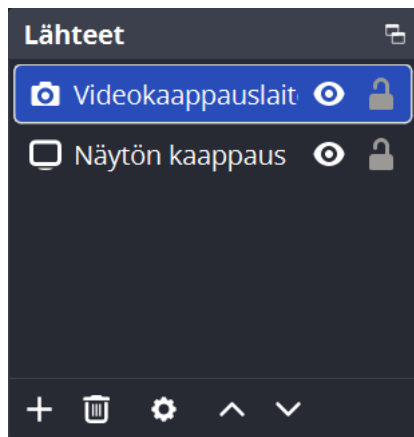


Kuva 10. Väriavain efektisuodattimen valitseminen



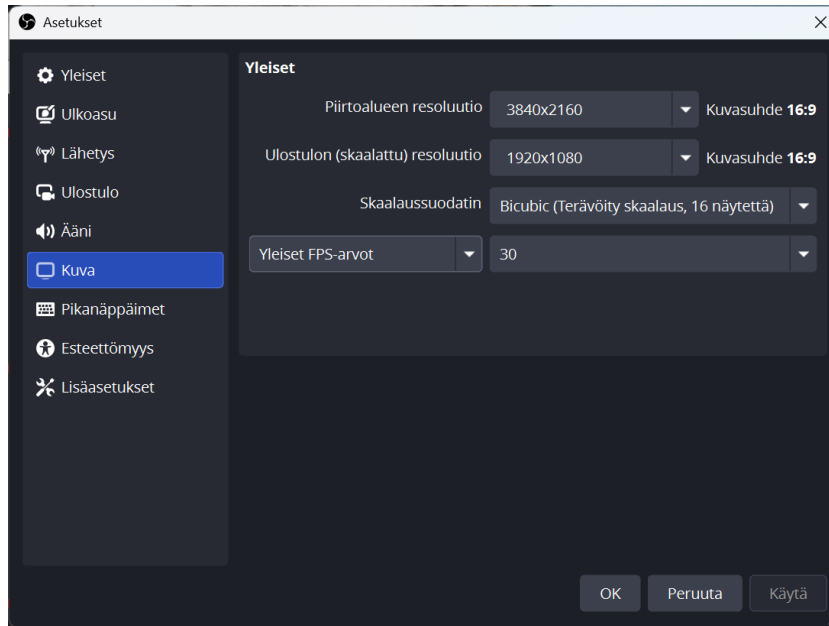
Kuva 11. Väriavaimen asetukset

Lopuksi on vielä tarkistettava, että näkymän lähteet on listattu oikeassa järjestyksessä. Videokaappauslaitte pitää olla listassa näytön kaappauksen yläpuolella (Kuva 12.), jotta videotallenteessa näkyy katseenseurantapiste heijastettuna katseltavan kohteen päällä. Kun näkymä on luotu oikein, katseenseurantapiste näkyy vain tallenteessa, eikä se näy ollenkaan testihenkilölle testitilanteen aikana.



Kuva 12. Lähteiden järjestys

Tässä kohdassa voi vielä tarkistaa OBS:n yleiset tallennuksen laadun tallennusasetukset. Voidaan valita matalampi ulostulon resoluutio ja yleinen FPS-arvo. Tässä työssä käytetään ruudun tallentamiseen 1920 x 1080p resoluutiota ja 30 Hz:n videokuvataajuutta, natiivin 3840 x 2160p resoluution ja 60 Hz:n ruudunpäivitystaajuuden sijasta (Kuva 13.) Jos käytössä on tehokkaampi tietokone, voidaan harkita suuremman resoluution tai suuremman videokuvataajuuden käyttöä (Cloudinary, 2025).



Kuva 13. Yleiset kuva asetukset

3.8 Inquisit 7 Lab -ohjelmistoympäristö

Inquisit 7 Lab (Inquisit) on lisenssityökalu psykologisten testien ja kokeiden suunnitteluun ja suorittamiseen (Millisecond, ei pvm.-a). Tähän työkaluun on tarjolla valmiita testejä ja testien ohjelmistokoodi on saatavilla muokattavaksi samalla lisenssillä (Millisecond, ei pvm.-b). Inquisit käyttää omaa Inquisit ohjelmointikieltä testien ohjelmointiin (Draine, 2022).

Inquisit on valittu tähän opinnäytetyöhön kuuluvaksi, koska Inquisit 7 versio tukee LSL -ominaisuutta. Turun ammattikorkeakoululla on käytössään laaja Inquisit-lisenssi, minkä vuoksi ohjelmiston soveltuvuutta ja käyttömahdollisuuksia halutaan tarkastella lisenssin tehokkaamman hyödyntämisen näkökulmasta.

Tässä työssä käytetään valmista Inquisit-testiä: Tower of London Task (ToL) (Millisecond, 2025). Työssä ei kuitenkaan perehdytä alkuperäisen testin tarkoitukseen, vaan perehdytään, kuinka LSL -ominaisuuksia voidaan käyttää Inquisit-ohjelmointikielessä. Testi on valittu myös, koska se toimii sopivana virikkeenä katseenseurantaa varten.

Kyseinen Tower of London Task, ladattuna Millisecond testikirjastosta sisältää 3 ohjelmakoodi-tiedostoa (Millisecond, 2025):

- toweroflondon.iqjs
- toweroflondon_inc.iqjs
- toweroflondon_instructions_inc.iqjs

Kaikki tarvittavat ohjelmakoodi muutokset tehdään tiedostoon:

toweroflondon_inc.iqjs ja kooditiedosto on kokonaisuudessaan liitteessä 1.

Tämä tiedosto sisältää pääasiallisen ohjelmistokoodin.

Inquisit-ohjelmistokoodiin voidaan lisätä toimintoja, jotka lähettävät erilaisia signaaleja, kun ohjelmassa tapahtuu jotain, mitä halutaan seurata testissä.

Tässä työssä lisätään toiminto, joka lähettää signaalin LSL -datavirtaan, kun testi aloitetaan ja kun testi päättyy. Mahdollisia signaaleja voi lisätä myös muihin testissä tapahtuviin kohtiin, mitä halutaankaan seurata kyseisessä testissä.

Alkuperäiseen defaults-koodilohkoon (Ohjelma 2.) lisätään LSL-portit (Ohjelma 3.):

Ohjelma 2. Alkuperäinen ohjelmakoodi.

```
404 = <defaults>
405 / canvasSize = (100%, 100%)
406 / canvasAspectRatio = (4,3)
407 / minimumVersion = "7.0.8.0"
408 / inputDevice = mouse
409 / screenColor = black
410 / fontStyle = ("Arial", 2.50%, true)
411 / txColor = white
412 / txBGColor = black
413 </defaults>
```

Ohjelma 3. Ohjelmakoodiin lisätty LSL-portit.

```

404 = <defaults>
405   / canvasSize = (100%, 100%)
406   / canvasAspectRatio = (4,3)
407   / minimumVersion = "7.0.8.0"
408   / inputDevice = mouse
409   / screenColor = black
410   / fontStyle = ("Arial", 2.50%, true)
411   / txColor = white
412   / txBGColor = black
413 </defaults>
414 = <port lsl_start>
415   / port = lsl
416   / items = (10)
417 </port>
418 = <port lsl_end>
419   / port = lsl
420   / items = (20)
421 </port>

```

Seuraavaksi lisätään uusi block-lohko:

Ohjelma 4. Uusi block-lohko.

```

1153 = <block startMarker>
1154   / trials = [1=startLSL]
1155
1156 </block>

```

Lisäksi lisätään kokonaan uusi trial-lohko, valmiina olevien trial-lohkojen väliin:

Ohjelma 5. Uusi trial-lohko

```

1075 *****
1076   Start Trial
1077 *****
1078 = <trial startLSL>
1079   / stimulusFrames = [1 = lsl_start]
1080   / trialDuration = 20
1081   / recordData = false
1082 </trial>

```

Tämä toiminto lähettää LSL-datavirtaan merkin "10", kun testi käynnistetään.

Lopuksi lisätään vielä valmiiseen trial finish-lohkoon (Ohjelma 6.) lsl_end, stimulusFrames- listaan (Ohjelma 7.)

Ohjelma 6. Alkuperäinen trial finish-lohko

```
1141 = <trial finish>
1142 / inputDevice = mouse
1143 / stimulusFrames = [1 = clearScreen, finish, finishInstruct, exitButton]
1144 / validResponse = (exitButton)
1145 / recordData = false
1146 </trial>
```

Ohjelma 7. Muokattu trial finish-lohko

```
1141 = <trial finish>
1142 / inputDevice = mouse
1143 / stimulusFrames = [1 = clearScreen, finish, lsl_end, finishInstruct, exitButton]
1144 / validResponse = (exitButton)
1145 / recordData = false
1146 </trial>
```

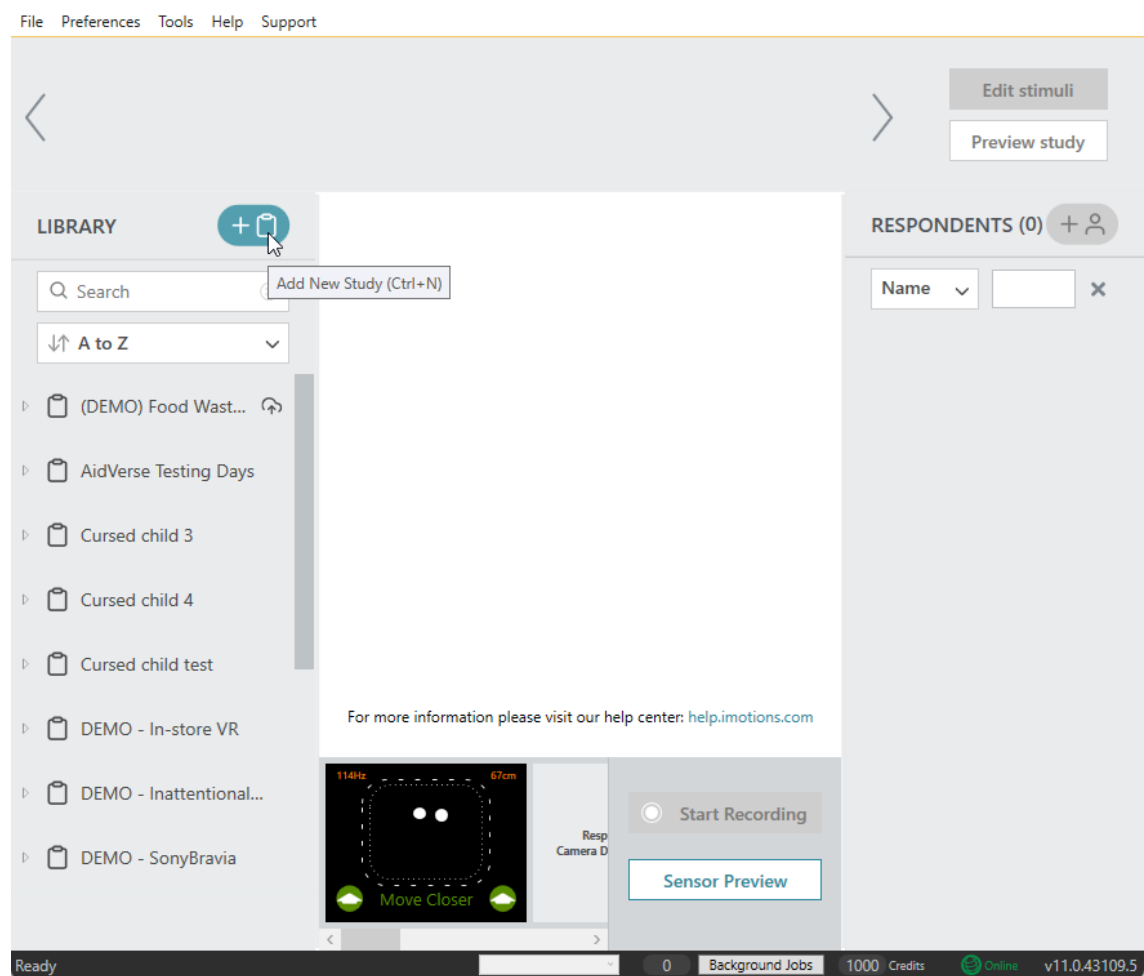
Tämä toiminto lähettää LSL-datavirtaan merkin "20", kun testi päättyy.

4 iMotions toteutus

Testin toteutus iMotions-ympäristössä aloitetaan luomalla testiympäristö iMotions-ohjelmaan ennen varsinaista testitilannetta.

4.1 Testiympäristön luominen iMotions -ympäristössä

Valitaan Add new Study (Kuva 14.).




Kuva 14. Uuden tutkimuksen luonti iMotions-ohjelmassa

Valitaan tutkimuksen tyypiksi Screen-based (Kuva 15.).


Create a new study

[Study builder help](#)

What kind of study are you planning to do?




Screen-based
Stimuli shown on a screen, (i.e. images, videos, websites, surveys, etc)




Glasses
Using eye tracking glasses


Choose this to create an experiment delivered on a computer screen in a lab.
Your study can contain images, videos, audio files, surveys, or recording the interaction with websites or other applications on the PC.




Screen-free recording
To record a scene or face, without screen-based stimuli



Mobile device
Mobile device screen with a mobile stand



Eye



VR

Cancel Next

Kuva 15. iMotions tutkimustyyppin valinta

Syötetään tarvittavat tiedot käytetystä näytöstä. (Kuva 16.)

Respondent screen

The size of the screen that will be used to display stimuli to responder

Why is this important? iMotions must know the physical size of your display in order to correctly calculate gaze angles from the raw eye tracking data.

Load from preset: TRT12

Resolution:

x

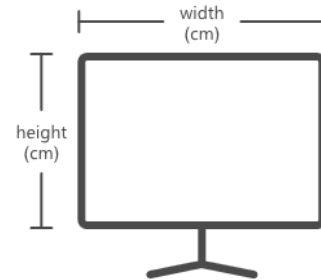
Width:

cm

Height:

cm

If you would like to save this as a custom size for future use, give it a name:



Kuva 16. iMotions tutkimuksessa käytettävän näytön tiedot

Valitaan tutkimuksessa käytettävät sensorit. Eye-tracking-valikosta valitaan käytössä oleva Tobii Pro Fusion-katseenseurantalaite. Valitaan mukaan Shimmer-laite. Valitaan Bluetooth Low Energy, jotta Polar H10-laite saadaan yhdistettyä. (Kuva 17.)

Select sensors

Which sensors do you want to use to collect data for this study?

The screenshot shows a 'Select sensors' interface. At the top, there are two selected sensor boxes: 'Eye tracking' (with a dropdown menu showing 'Tobii Pro Fusion') and 'Respondent camera' (with a checked checkbox for 'AFFDEX'). Below these are several other sensor options in a grid: 'ActiCHamp', 'Biopac', 'Demo', 'Emotiv Cortex', 'Lab Streaming Layer', 'Shimmer', and 'Bluetooth Low Energy'. A vertical scrollbar is visible on the right side of the grid.

Note that selecting the sensors for this study means that iMotions will collect data for these sensors, if available. You must also make sure each of the selected sensors is connected and configured in [Global Settings](#).

Back

Next

Kuva 17. iMotions tutkimuksessa käytettävien sensoreiden valinta

Annetaan tutkimukselle nimi. (Kuva 18.)

Name your study

What do you want your study to be called?

Opinnäytetyö esimerkki

Use the text area below to add notes for your study

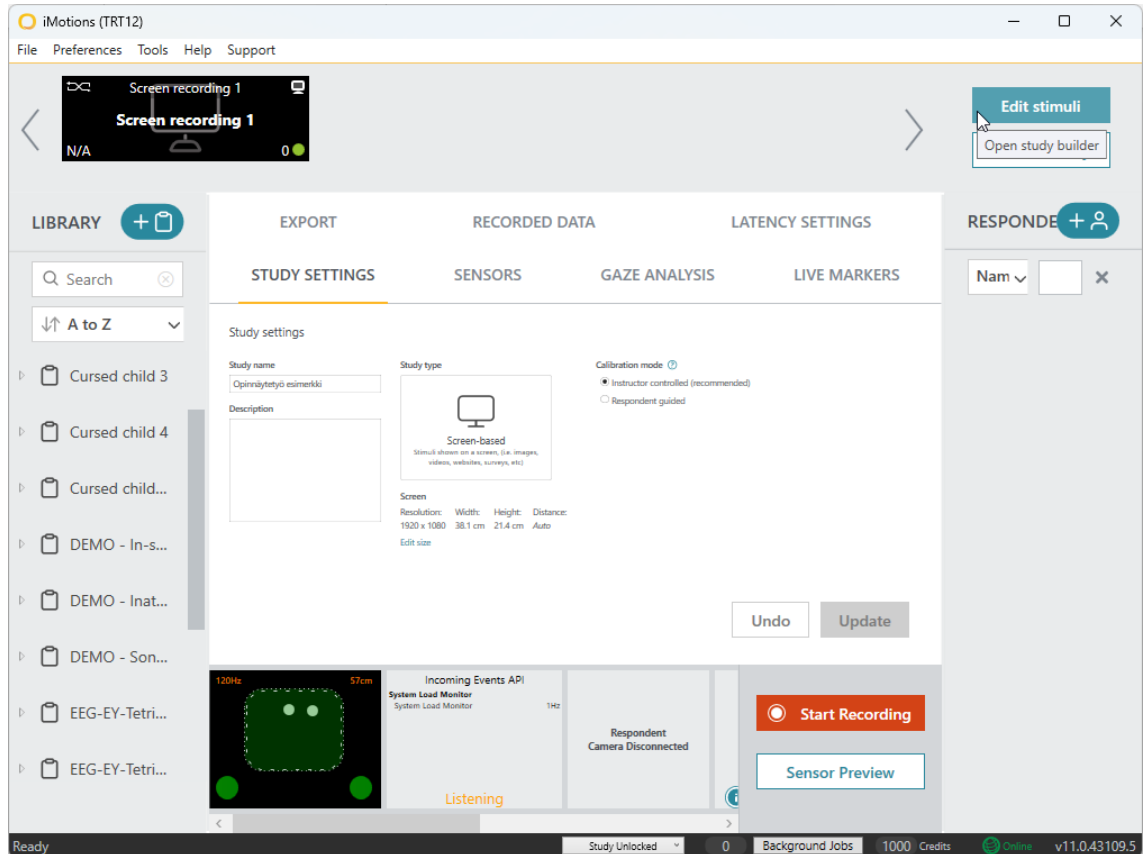
The screenshot shows a 'Name your study' interface. It features a text input field with the placeholder text 'Opinnäytetyö esimerkki' and a large, empty text area below it for adding notes.

Back

Create study

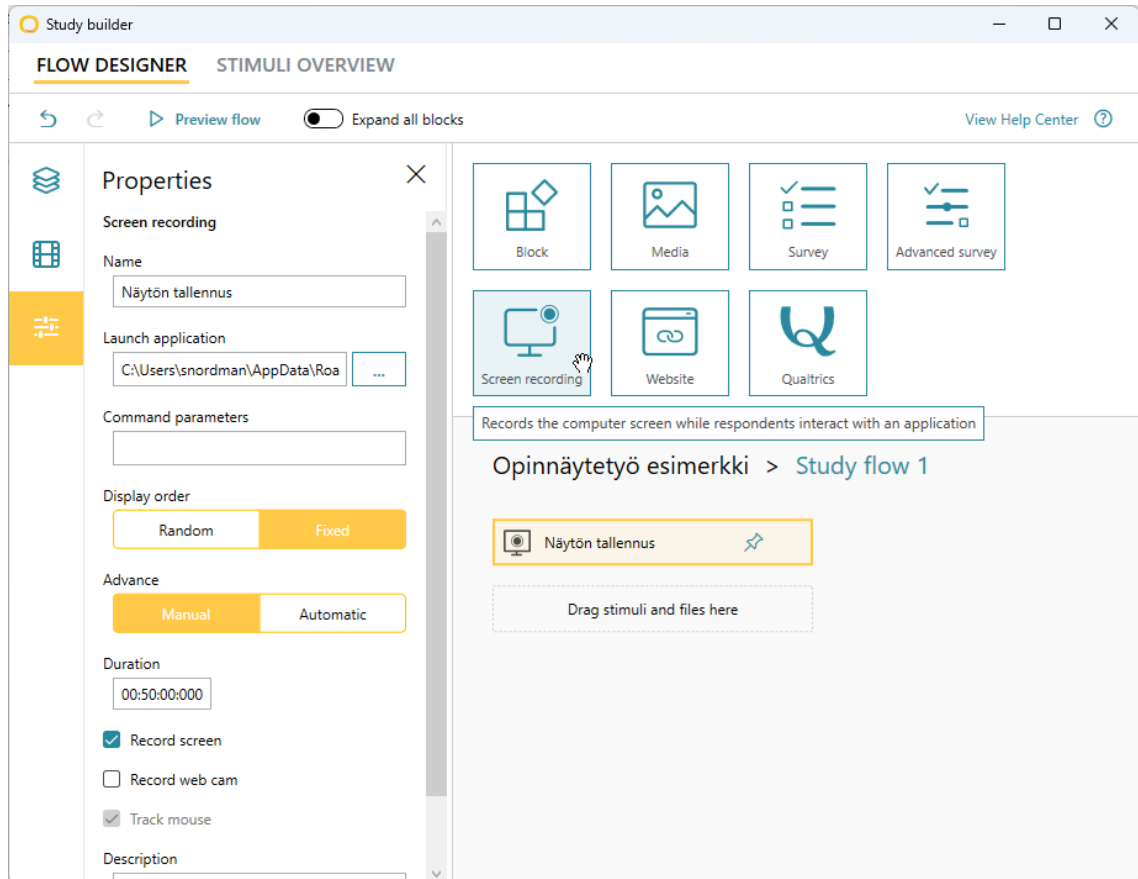
Kuva 18. iMotions tutkimuksen nimeäminen

Tutkimukseen lisätään vielä jokin virike, jota halutaan tutkia. Tässä esimerkissä käytetään Inquisit-ohjelmaa testin suorittamiseen. iMotions-tutkimuksessa valitaan Edit stimuli (Kuva 19.).



Kuva 19. Virikkeen lisääminen iMotions-ympäristöön

Valitaan Screen recording, asetetaan sopiva nimi ja valitaan Launch application Inquisit-sovelluksen tiedostosijainti. Valitaan Record screen -asetus aktiiviseksi (Kuva 20.). Launch application -valinnassa voidaan käyttää myös jotain muuta sovellusta, mitä halutaan käyttää testissä.



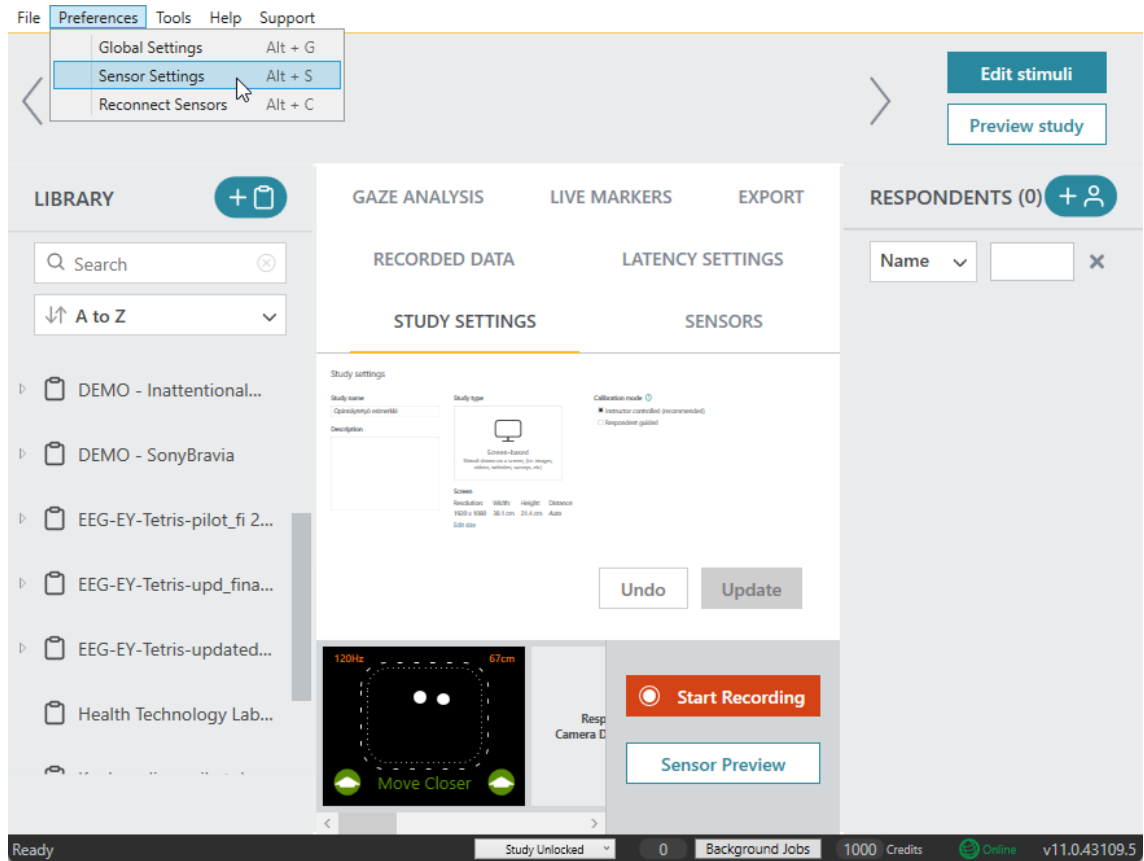
Kuva 20. iMotions-ympäristön virikkeen asetukset

4.2 Laitteiden yhdistäminen iMotions-ympäristöön

Tutkimuksessa käytettävät laitteet yhdistetään iMotions-ympäristöön seuraavasti.

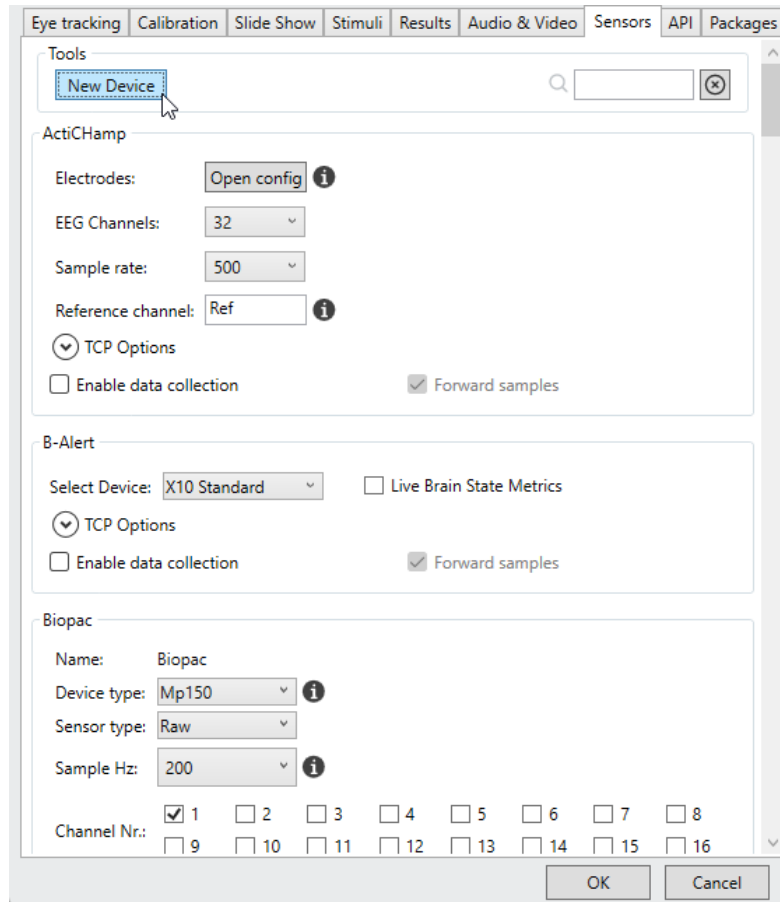
4.2.1 Polar H10-laitteen yhdistäminen

1. Polar H10 -laitteen rintavyön elektrodialue kostutetaan vedellä ja kiinnitetään testihenkilön rintakehälle ihoa vasten.
2. Päälaite kiinnitetään rintavyön napoihin.
3. Laite yhdistetään Bluetooth-yhteydellä tietokoneeseen.
4. Laite lisätään iMotions-ohjelmassa valitsemalla Preferences ja Sensor Settings (Kuva 21.).



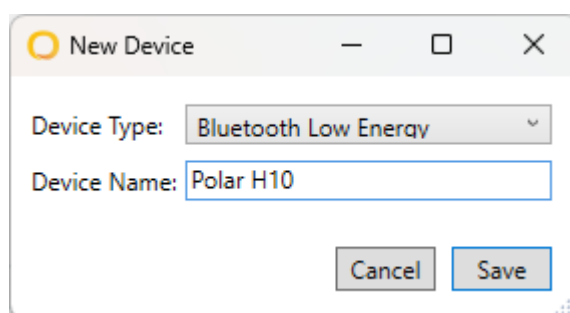
Kuva 21. Sensori asetukset iMotions-ohjelmassa

5. Valitaan New Device (Kuva 22.).



Kuva 22. Uuden laitteen lisääminen iMotions-ohjelmaan

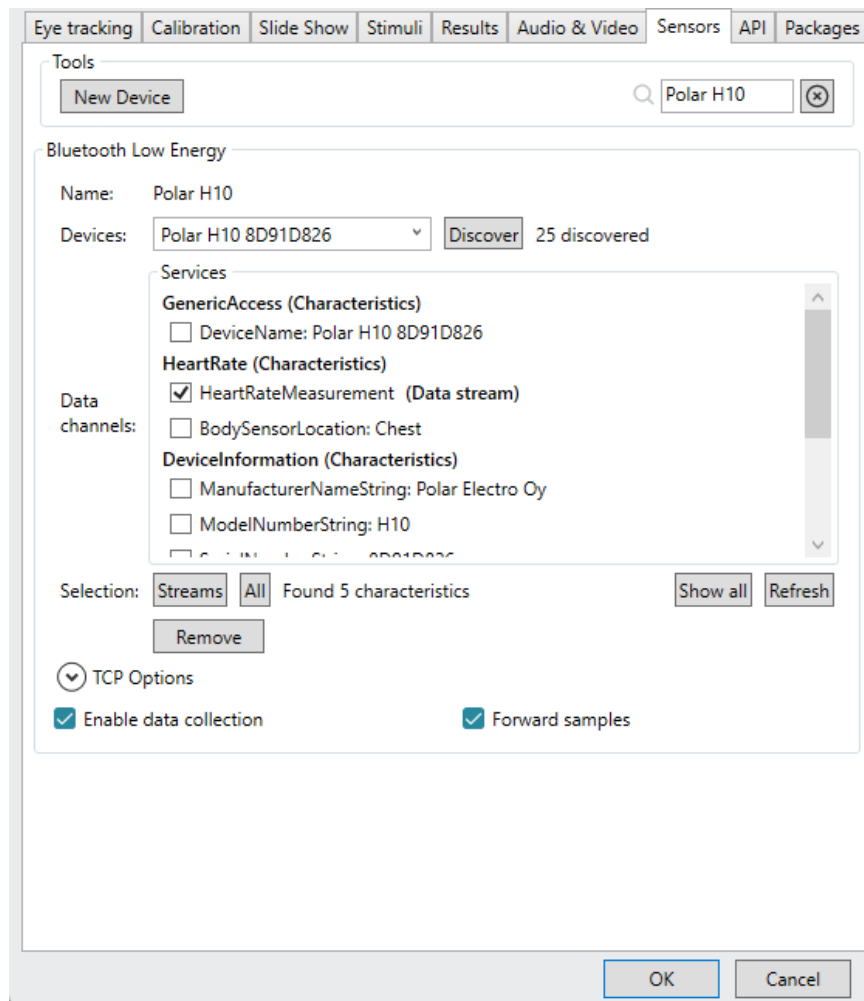
6. Valitaan laittaan Device type valikosta Low Energy Bluetooth ja asetetaan Device Name - kenttään laitteen nimi (Kuva 23.).



Kuva 23. Lisättävän laitteen tyyppi ja nimi

7. Seuraavassa näkymässä valitaan Discover ja valitaan valikosta oikea Polar H10 -laite sen sarjanumeron perusteella, joka löytyy laitteen pääyksiköstä. Valitaan Data channels -valikosta HeartRateMeasurement

ja TCP Options -valikosta Enable data collection sekä Forward samples -valinnat. (Kuva 24.).

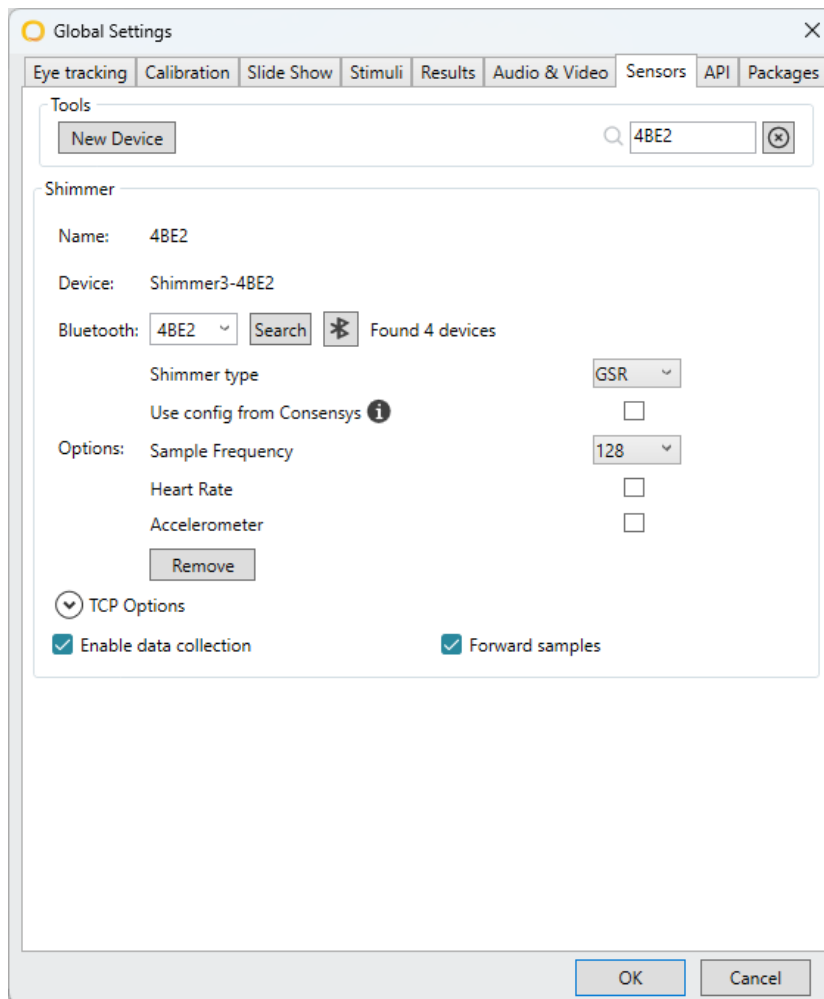


Kuva 24. Polar H10 -laitteen asetukset

4.2.2 Shimmer3 GSR+ -laitteen yhdistäminen

1. Kiinnitetään rannehihna testihenkilön ranteeseen.
2. Kiinnitetään kuivaelektrodit sormiin.
3. Käynnistetään laitteen virta.
4. Yhdistetään laite Bluetooth-yhteydellä tietokoneeseen.

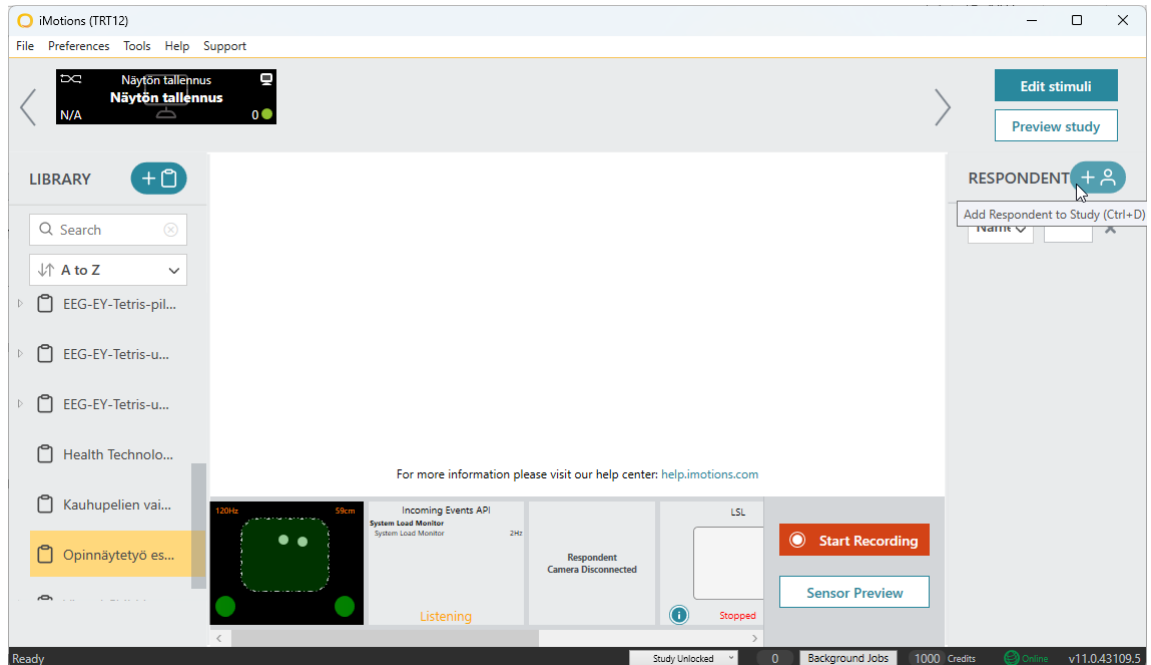
5. Laitteen yhdistäminen iMotions-ohjelmaan tapahtuu samalla tavalla kuin Polar H10-laitteen lisääminen, muutamalla eroavaisuudella. Laitteen tyyppiä valitaan "Shimmer" ja nimenä käytetään laitteen sarjanumeroa, joka löytyy laitteesta.
6. Valitaan laitteen sarjanumero listasta ja valitaan Shimmer type: GSR (Kuva 25.).
7. Käytetään samoja TPC-valintoja, kuin Polar H10 laitteessa (Kuva 25.).



Kuva 25. Shimmer-laitteen asetukset

4.3 Testin suoritus ja tallentaminen

Ennen testin suorittamista, tarvitsee iMotions-tutkimukseen lisätä Responder (Kuva 26.).



Kuva 26. Vastaajan lisääminen iMotions-tutkimukseen

Vastaajan tiedot voidaan täyttää haluamalla tavalla, riippuen millaista tutkimusta ollaan tekemässä (Kuva 27.).

Name

Gender Age

Group

Comments

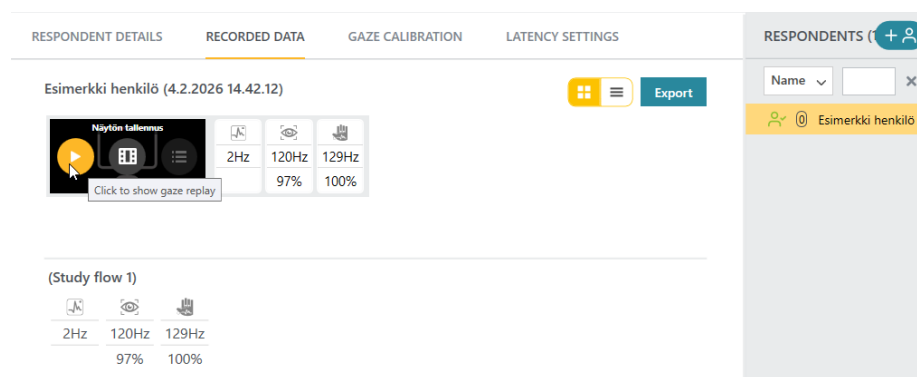
Kuva 27. Vastaajan tiedot

Kun vastaajan tiedot on asetettu, voidaan valita vastaaja ja aloittaa tallennus Start Recording -painikkeella.

iMotions aloittaa katseenseurannan kalibroinnin, jossa testihenkilön on seurattava katseellaan näytölle ilmaantuvia pisteitä. Onnistuneen kalibroinnin jälkeen iMotions käynnistää Inquisit-sovelluksen ja varsinainen testin suoritus aloitetaan. Suoritettuaan testin voi testihenkilö sulkea Inquisit-sovelluksen, jolloin iMotions lopettaa tallentamisen.

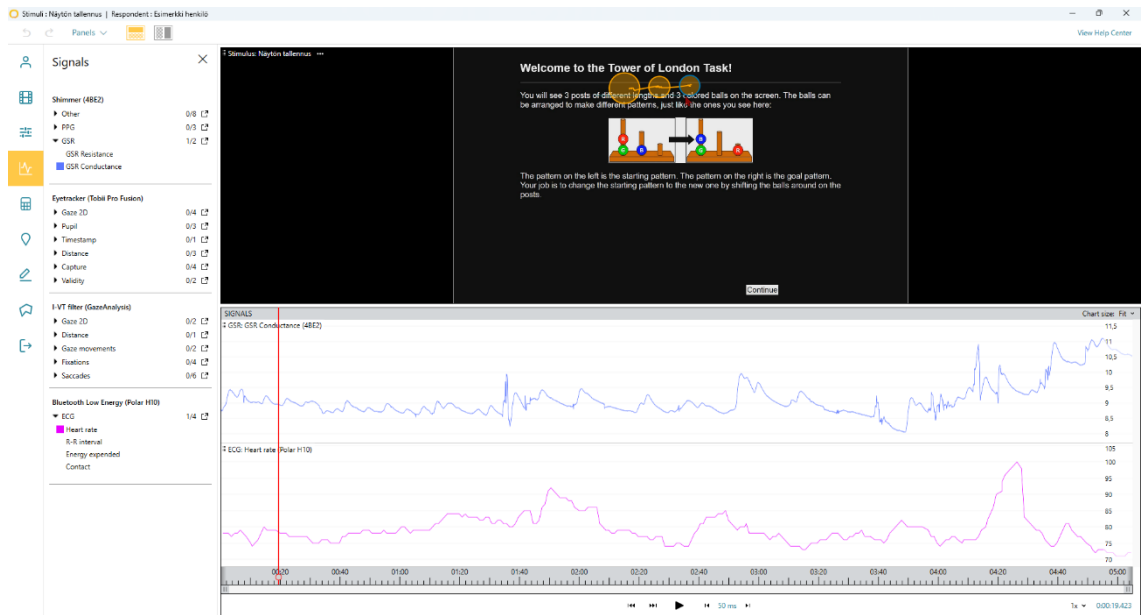
4.4 Tallennettujen tietojen käyttö iMotions-ympäristössä

Tallennettuja tietoja voidaan tutkia valitsemalla haluttu vastaaja ja avaamalla tallenteen toisto (Kuva 28.).



Kuva 28. Tallenteen avaaminen iMotions-ohjelmassa

Avatussa tallenteen näkymässä voidaan valita Signals-valikosta kaikki signaalit, joita halutaan seurata. Näkyväksi tiedoiksi on valittu Shimmer-laitteesta GSR sähkönsäilytys ja Polar H10-laitteesta pulssilukema. Valittujen signaalien arvot piirtyvät kuvioon aikajanelle ja ruudunkaappaustallenne näkyy katseenseurantapisteenä tässä näkymässä. iMotions näyttää viimeisimmät katseen kohdepisteen ympyröinä, jotka kuvaavat kohdepisteen kestoa. Isompi ympyrä tarkoittaa pidempää katseen kohdistuksen kestoa (Kuva 29.). Tässä esimerkissä on käytetty käyttöjärjestelmän kursoria kiintopisteenä, jota seurattiin tallennuksen aikana, jotta katseenseurannan tarkkuutta voitaisiin arvioida.



Kuva 29. iMotions-tallenteen näkymä

iMotions-tallennetuista sensorisignaaleista voidaan luoda CSV-tiedosto ulkoista analyysiä varten (iMotions, ei pvm.-a). CSV-tiedoston käyttö ulkoisessa analyysissä ei kuulu tämän opinnäytetyön laajuuteen.

5 Uuden ratkaisun toteutus

Tässä luvussa kuvataan uusi mittausratkaisu, jota verrataan vaihtoehtoina iMotions-pohjaiselle toteutukselle. Uusi ratkaisu koostuu modulaarisista komponenteista, joissa fysiologiset mittaukset ja katseenseuranta yhdistetään Lab Streaming Layer (LSL) -väliohjelmiston avulla. Ratkaisun tavoitteena on mahdollistaa mittausdatan synkronointi ja tallennus myöhempää analyysiä varten kustannustehokkaasti.

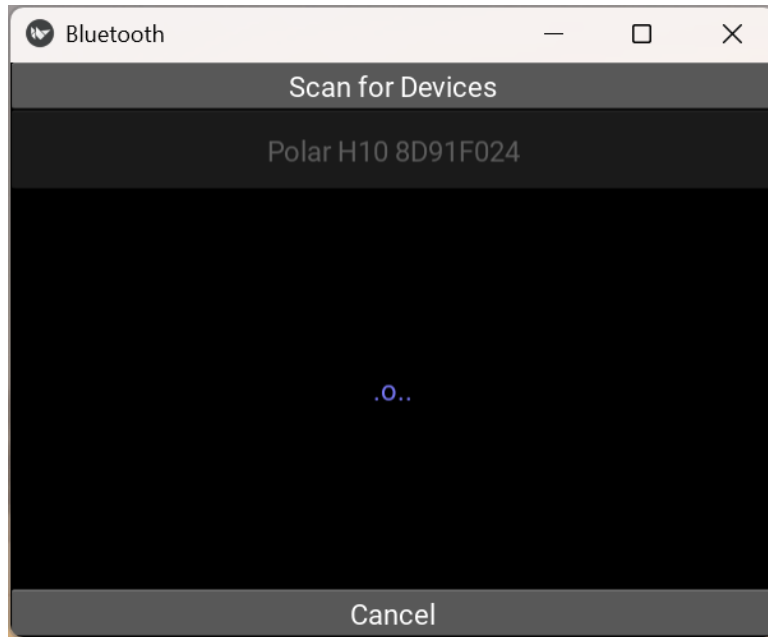
5.1 Uuden ratkaisun komponentit

Toteutus koostuu erillisistä laite- ja ohjelmistokomponenteista, jotka kootaan toimivaksi kokonaisuudeksi. Luvussa esitellään ensin käytetyt laitteet ja niihin liittyvät ohjelmistot, minkä jälkeen kuvataan datavirran muodostaminen, mittausdatan tallennus sekä toteutuksen keskeiset tekniset ratkaisut.

Käytännön toteutus edellyttää vaiheittain etenemistä. Tarvittavat ohjelmat ja ohjelmistot pitää olla valmiiksi konfiguroitu ennen tallennus tilanteen aloittamista.

5.1.1 Polar H10 -laitteen yhdistäminen LSL -datavirtaan

1. Polar H10-laitteen rintavyön elektrodialue kostutetaan vedellä ja kiinnitetään testihenkilön rintakehälle ihoa vasten.
2. Päälaite kiinnitetään rintavyön napoihin.
3. Laite yhdistetään Bluetooth-yhteydellä tietokoneeseen.
4. Käynnistetään PolarBand2Isl-sovellus.
5. Laite yhdistetään sovellukseen.
6. Varmistetaan, että yhteys on muodostunut. Yhteys on muodostunut, kun sovelluksessa näkyy liikkuva o-merkki janalla (Kuva 30.).



Kuva 30. Toimiva yhteys PolarH10-laitteeseen

5.1.2 Shimmer3 GSR+-laitteen yhdistäminen LSL-datavirtaan

1. Kiinnitetään rannehihna testihenkilön ranteeseen ja kiinnitetään kuivaelektrodit sormiin.
2. Käynnistetään laitteen virta.
3. Yhdistetään laite Bluetooth-yhteydellä tietokoneeseen.
4. Käynnistetään ShimmerPPGHRGSRConsoleAppExample-sovellus.
5. Varmistetaan, että yhteys on muodostunut. Yhteyden voi varmistaa seuraamalla sovelluksen käynnistämää komentokehotetta ja komentokehotteeseen tulostuu uudet mitatut arvot sekunnin välein (Kuva 31.).

```

ACK Received
ACK Received
ACK Received
Connected on COM4
ACK Received
Shimmer is Streaming
ACK for Streaming Command Received
AccelX: 24,4891304347826 m/(sec^2) AccelY: 24,4891304347826 m/(sec^2) AccelZ: 24,4891304347826 m/(sec^2)
Time Stamp: 1770709355578,54 mSecs GSR: -40,2 kOhms PPG: 0 mVolts HR: -1 BPM
AccelX: 5,22826086956522 m/(sec^2) AccelY: 2,21739130434783 m/(sec^2) AccelZ: 10,554347826087 m/(sec^2)
Time Stamp: 1770709356578,57 mSecs GSR: 4504500,00000002 kOhms PPG: 1050,54945054945 mVolts HR: -1 BPM
AccelX: 5,23913043478261 m/(sec^2) AccelY: 2,18478260869565 m/(sec^2) AccelZ: 10,5108695652174 m/(sec^2)
Time Stamp: 1770709357578,57 mSecs GSR: 4504500,00000002 kOhms PPG: 1059,34065934066 mVolts HR: -1 BPM

```

Kuva 31. Shimmer C# Api komentokehote

5.1.3 EyeTrax- virtuaalikameran käynnistäminen

1. Käynnistetään sovellus komentoriviltä sen tiedostosijainnista ja käytetään asetus komentoja (Ohjelma 8.).

Ohjelma 8. EyeTrax-sovelluksen käynnistäminen komentoriviltä

```

\Desktop\eyetrax_test> eyetrax-virtualcam --filter kalman
--calibration 5p

```

2. Suoritetaan katseen kalibrointi.
3. Tässä kohtaa, testihenkilön tulisi katsoa vain näyttöä, jotta kalibrointi pysyy optimaalisena.

Ohjelma on oikein käynnistetty, kun komentokehoteessa näkyy ohjelman lähettämä viesti: Virtual camera started: OBS Virtual Camera.

5.1.4 OBS videotallenteen aloitus

Oikeanlainen OBS näkymä tulisi olla valmiiksi luotuna, ennen kuin tallennus aloitetaan.

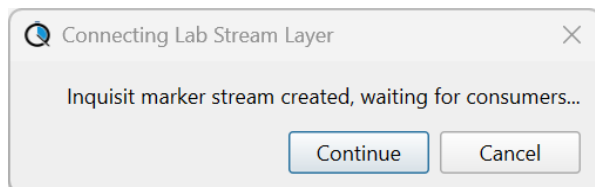
1. Varmistetaan, että näkymä on rakennettu oikein.

2. Aloitetaan videotallennus.

5.1.5 Inquisit

Inquisit- ohjelmisto toimii tässä työssä virikkeen tuottajana. Ratkaisu ei kuitenkaan ole sidottu kyseiseen ohjelmistoon, vaan sen tilalla voidaan käyttää mitä tahansa virikkeitä tuottavaa sovellusta, joka tukee tarvittavaa LSL-yhteensopivuutta.

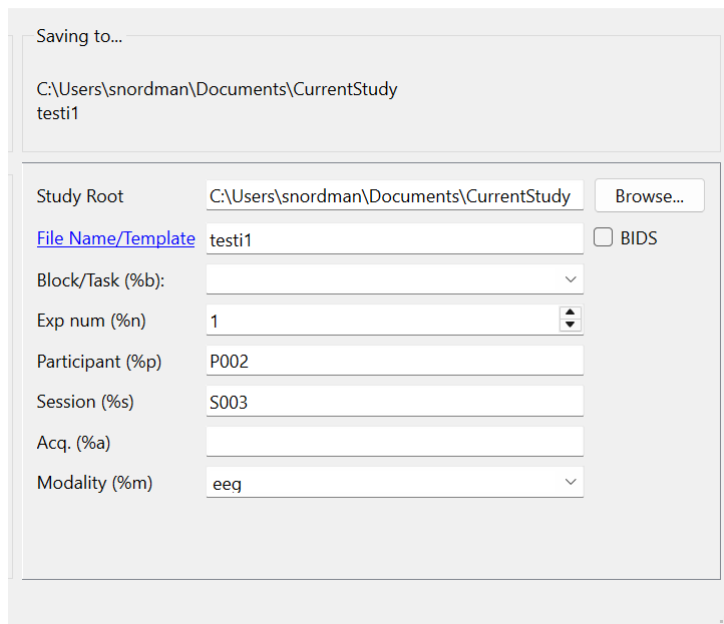
1. Käynnistetään haluttu testi.
2. Jos testiin on ohjelmoitu LSL-toimintoja, tässä kohtaa Inquisit jää odottamaan, kunnes LSL -datavirran tallennus aloitetaan toisessa ohjelmassa (Kuva 32.). Tässä työssä LSL -datavirran tallentamiseen käytetään Lab Recorder -sovellusta. Ponnahdusikkuna katoaa, kun LSL -tallennus aloitetaan. On myös tärkeää, että ponnahdusikkunaa ei suljeta valitsemalla Continue. Tämä aloittaa testin ilman, että LSL -datavirtaa tallennetaan.



Kuva 32. Inquisit odottaa LSL-tallentajaa

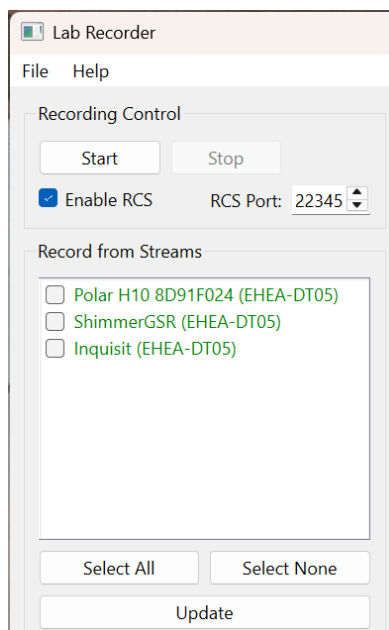
5.1.6 Lab Recorder

Ennen testilanteen aloittamista, on hyvä tarkistaa Lab Recorder -sovelluksen tiedot, minne tiedostoon tallennus luodaan ja tarkistaa muut tallennukseen liittyvät metatiedot. File Name/Template -kenttään voidaan syöttää millä nimellä tiedosto tallennetaan Study Root valittuun tiedostosijaintiin (Kuva 33.). Tiedoston nimeämisessä voidaan halutessa käyttää %-parametreja tietojen modulaariseen täyttöön. Tämän ominaisuuden käyttö ei kuitenkaan ole pakollista.



Kuva 33. Lab Recorder tallennustietojen syöttäminen

1. Tarkistetaan, että kaikista tarvittavista lähteistä havaitaan datavirrat. Tässä esimerkissä havaitaan Polar H10, ShimmerGSR ja Inquisit datavirrat (Kuva 34.).



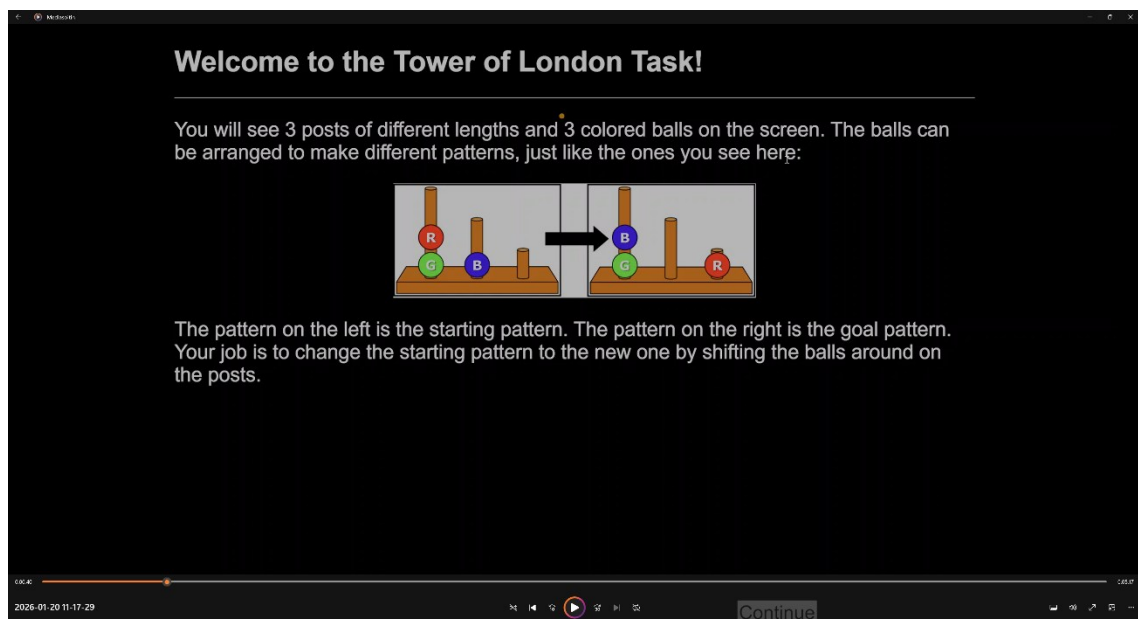
Kuva 34. Datavirrat havaittuna Lab Recorder-sovelluksessa

2. Valitaan kaikki datavirrat ja aloitetaan tallennus.
3. Inquisit testi käynnistyy nyt varsinaisesti.
4. Kun Inquisit testi on saatu päätökseen. Lopetetaan LSL -tallennus Lab Recorder -sovelluksesta.
5. Lopetetaan OBS -videotallennus.

5.2 Tallennettujen tiedostojen käyttö

5.2.1 OBS -videotallenne

OBS luoman videotallenteen voi löytää avaamalla OBS-sovelluksesta Tiedosto ja Näytä tallenteet. Oletuksena tallenteet tallentuvat käyttöjärjestelmän Videot-kansioon ja videot nimetään aikaleimalla, jolloin tallennus on aloitettu. Videotallenteessa näkyy ruutu, jota testihenkilö näkee ja videotallenteessa näkyy oranssi katseenseurantapiste kullakin hetkellä (Kuva 35.).

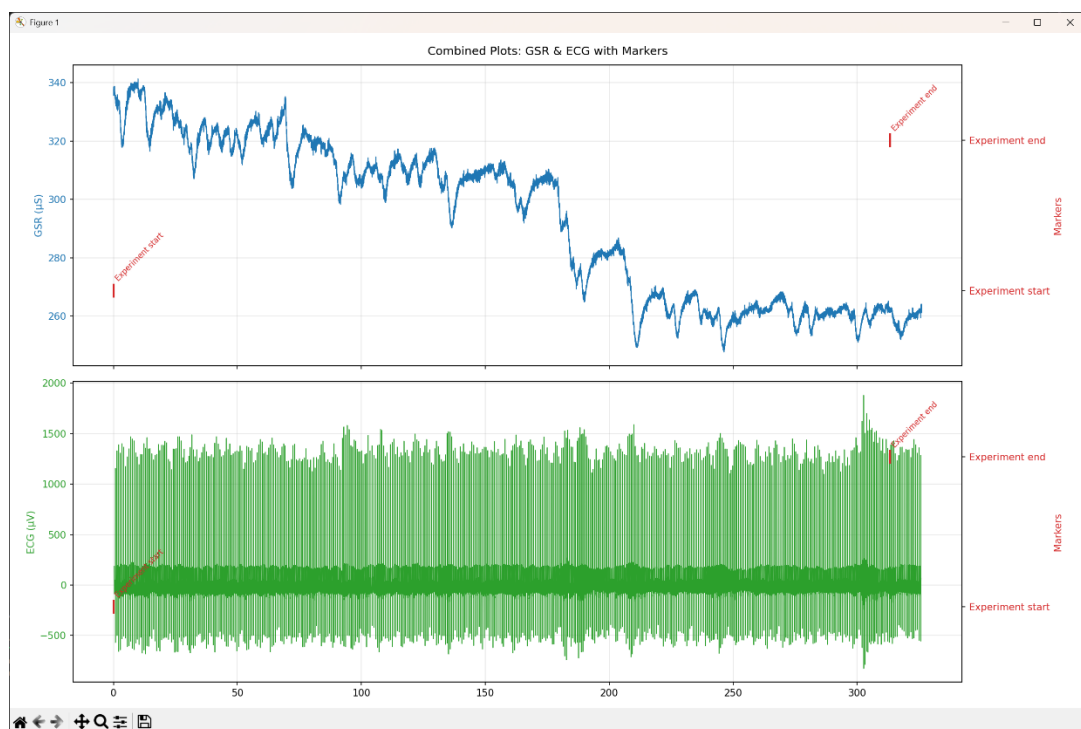


Kuva 35. OBS -videotallenne

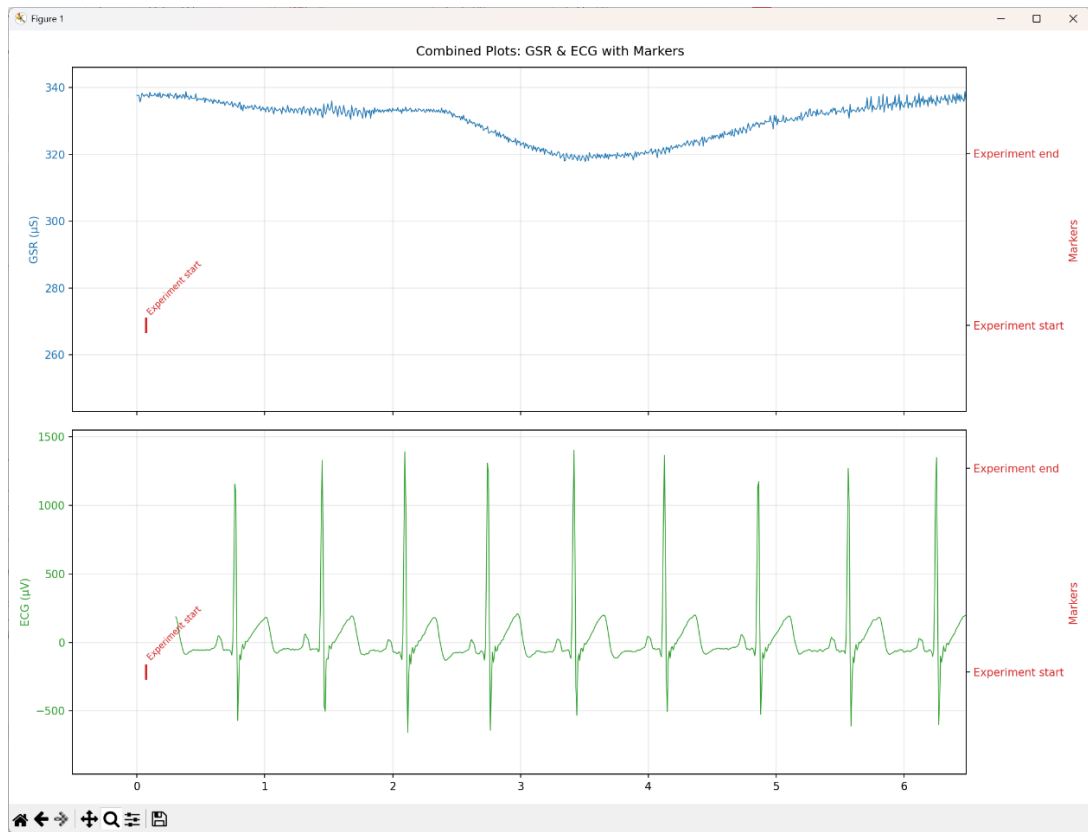
5.2.2 Lab Recorder XDF -tiedosto

XDF-tiedoston käsittelyyn tässä työssä käytetään PyXDF-Pythonkirjastoa. Kirjastoa tarvitaan, jotta XDF-tiedostoja voidaan ladata oikein muuta käyttöä varten (xdf-modules, 2026). Tietojen esittämiseen hyödynnetään NumPy ja Matplotlib-kirjastoja. Ohjelmistokoodiin muokataan oikea tiedostopolku, josta löytyy haluttu XDF-tiedosta tarkastelua varten. Käytetty ohjelmakoodi kokonaisuudessaan löytyy liitteestä 1.

Ohjelmakoodi piirtää kuvion, joka sisältää 2 alakuviota. Kuviossa näkyy mitatut GSR ja EKG -arvot aikajanalla, sekä merkit, jolloin testi aloitettiin ja merkki, jolloin testi päättyi (Kuvio 1.). Kuviota voidaan myös zoomata, jos halutaan tarkastella jotain tiettyä aikaväliä, tai jos halutaan nähdä tarkemmin osa kuviosta (Kuvio 2.).



Kuvio 1. Yhdistetty kuvio



Kuvio 2. Zoomattu kuvio

6 Tulosten vertailu

6.1 Vertailuasetelma ja arviointiperusteet

Tässä luvussa verrataan iMotions-pohjaista mittausratkaisua ja tässä työssä luotua ratkaisua. Vertailun tavoitteena ei ole analysoida mitattujen fysiologisten arvojen merkitystä, vaan tarkastella soveltuvuutta opetuskäyttöön. Arviointi perustuu tämän opinnäytetyön aikana tehtyihin mittauksiin sekä tekijän tekemiin havaintoihin.

Vertailu perustuu seuraaviin perusteisiin:

- mittausten laatu ja luotettavuus
- datan synkronointi ja jatkokäytettävyys
- kustannukset ja skaalautuvuus
- soveltuvuus opetuskäyttöön

6.2 Mittausdatan laatu

iMotions-pohjaisessa ratkaisussa katseenseurannassa käytettiin Tobii Pro Fusion -laitetta. Testien perusteella tarkkuus oli eriomainen ja seurasi hyvin nopeasti liikkuvaa katsetta.

Uudessa ratkaisussa katseenseurantaan käytettiin verkkokameraa hyödyntävää EyeTrax-ohjelmaa. Testien perusteella katseenseurannan tarkkuus oli selkeästi heikompi, etenkin näytön reunoja lähestyessä ja seuranta oli hidasta nopean katseen tulkinnassa. Kalibrointi oli myös herkempi pään liikkeille, johtuen heikompaan tarkkuuteen nopeammin.

Katseenseurannan osalta voidaan todeta, että uuden ratkaisun käyttämistä on harkittava sen perusteella, kuinka tarkkoja tuloksia vaaditaan katseenseurannalta.

Molemmissa ratkaisuissa kerättiin fysiologista dataa käyttäen samoja laitteita. Kerätyissä GSR-arvoissa ei havaittu merkittäviä laadullisia eroja.

iMotions-ympäristössä kerätty EKG-data automaattisesti käsiteltiin ja esitettiin suoraan käyttäjäystävällisenä sykelukuna.

Uudessa ratkaisussa PolarBand2Isl-ohjelma lähettää mitatut EKG-arvot raakadatana ja sykkeen esittäminen lukuarvona vaatisi jatkokäsittelyä.

Raakadatatasta pystytään kuitenkin piirtämään sydänsähkökäyrä, jota voidaan käyttää muuhun tarkoitukseen.

6.3 Datan synkronointi ja jatkokäytettävyys

iMotions-ympäristössä kaikki mitatut arvot ja videotallenne on automaattisesti synkronoitu ja tuloksia on helppo tarkastella iMotions-ohjelmistossa. Lisäksi mahdollisuus viedä tallennetut tiedot CSV-tiedostona mahdollistaa datan käytön ulkoisissa ympäristöissä.

Uudessa ratkaisussa LSL-datavirrat on luotettavasti synkronoitu ongelmitta. Videotallenne on kuitenkin erillisessä ympäristössä mitattujen arvojen kanssa. Tämä hankaloittaa tarkkojen mitattujen arvojen synkronointia videotallenteeseen verraten.

6.4 Kustannukset ja skaalautuvuus

iMotions-ohjelman käyttö vaatii lisenssimaksuja ja on sidottu rajalliseen määrään laitteita (iMotions, 2023). Tämä rajoittaa samanaikaisten tutkimusten määrää.

Uusi ratkaisu perustuu avoimen lähdekoodin ohjelmistojen käyttöön eikä vaadi erillisiä lisenssimaksuja. Ratkaisun voisi toteuttaa millä tahansa tietokoneella, jossa on verkkokamera ja käyttöä rajoittaisi eniten Shimmer-laitteiden saatavuus. Tämä mahdollistaa useamman kokonaisuuden toteuttamisen samaan aikaan.

7 Lopuksi

Tämän opinnäytetyön tavoitteena oli selvittää, voidaanko Turun ammattikorkeakoulun INSIGHT Labissa käytössä olevaa iMotions-pohjaista mittausratkaisua osittain korvata kustannustehokkaammalla ja joustavammalla avoimeen lähdekoodiin perustuvalla vaihtoehdolla siten, että mittausdatan laatu ja käytettävyys säilyvät riittävinä.

Tulosten perusteella avoimen lähdekoodin ratkaisu ei täysin korvaa iMotions-järjestelmää, erityisesti katseenseurannan tarkkuuden ja käyttömukavuuden osalta. Verkkokamerapohjainen katseenseuranta oli tarkkuudeltaan heikompi eikä vastannut Tobii Pro Fusion -laitteiston tarkkuutta. Fysiologisten mittausten osalta tulokset olivat kuitenkin vertailukelpoisia. Polar H10- ja Shimmer3 GSR+ -laitteilla kerätty data saatiin synkronoitua ja tallennettua Lab Streaming Layer -väliohjelmiston avulla luotettavasti. LSL osoittautui toimivaksi ratkaisuksi datavirtojen yhdistämiseen ja se tukee modulaarista rakennetta.

Työssä onnistuttiin rakentamaan toimiva ja kustannustehokas mittauskokonaisuus. Ratkaisu soveltuu kevyempiin tutkimusasetelmiin. Haasteita ilmeni järjestelmän konfiguroinnissa ja käytettävyydessä, sillä uusi ratkaisu edellyttää useiden erillisten ohjelmistojen hallintaa ja lisää käyttäjän teknistä vastuuta verrattuna valmiiseen iMotions-ympäristöön.

Tuloksiin liittyy rajoituksia. Testaus toteutettiin pelkästään tekijän toimesta ja tulosten vertailu perustui pitkälti tekijän mielipiteisiin. Lisäksi toteutus testattiin vain yhdessä käyttöympäristössä. Näin ollen tulosten yleistettävyys on rajallinen.

Uutta ratkaisua voidaan hyödyntää tilanteissa, joissa kustannustehokkuus, laitteiston saatavuus ja modulaarisuus ovat keskeisiä. Tarkempia tuloksia vaativissa tutkimuksissa iMotions-ympäristö säilyttää etunsa. Jatkokehitys voisi kohdistua lisensoitujen verkkokamerapohjaisten katseenseurantaratkaisujen

tarkempaan vertailuun sekä erilaatuisten verkkokameroiden eroja katseenseurannassa.

Laajemmassa tarkastelussa työ osoittaa, että avoimeen lähdekoodiin perustuvat ratkaisut voivat täydentää lisensoituja tutkimusympäristöjä ja lisätä mittausteknologian saavutettavuutta. Vaikka uusi ratkaisu ei täysin korvaa iMotions-järjestelmää, sitä voidaan käyttää harkituissa käyttökohteissa.

Lähteet

- Ball, L., & Richardson, B. (2023). *Eye-Tracking and Physiological Measurements for UX Evaluation*. <https://doi.org/10.1201/9781003495161-9>
- Can, Y. S., Arnrich, B., & Ersoy, C. (2019). Stress detection in daily life scenarios using smart phones and wearable sensors: A survey. *Journal of Biomedical Informatics*, 92, 103139. <https://doi.org/10.1016/j.jbi.2019.103139>
- ck-zhang. (2025, huhtikuuta 27). *EyeTrax — Webcam-based Eye Tracking Library*. https://www.reddit.com/r/computervision/comments/1k8whbd/eyetrax_webcambased_eye_tracking_library/
- Cloudinary. (2025, marraskuuta 17). *Video Compression Techniques: Enhancing Quality and Performance*. <https://cloudinary.com/guides/video-effects/video-compression-techniques>
- Draine, S. (2022). *Inquisit Programmer's Manual*. <https://www.millisecond.com/support/Inquisit%20Programmer's%20Manual.pdf>
- iMotions. (ei pvm.-a). *Data Export*. Noudettu 11. helmikuuta 2026, osoitteesta <https://imotions.com/products/imotions-online/data-export/>
- iMotions. (ei pvm.-b). *iMotions Lab*. Noudettu 22. tammikuuta 2026, osoitteesta <https://imotions.com/products/imotions-lab/>
- iMotions. (ei pvm.-c). *Solutions - iMotions*. Noudettu 30. tammikuuta 2026, osoitteesta <https://imotions.com/solutions/>
- iMotions. (2023). *iMotions Software Module Descriptions*. https://imotions.com/wp-content/uploads/brochures/imotions_module_descriptions.pdf

INSIGHT Lab. (2025). *Healt Tech Lab's devices*.

Kothe, C., Shirazi, S. Y., Stenner, T., Medine, D., Boulay, C., Grivich, M. I., Artoni, F., Mullen, T., Delorme, A., & Makeig, S. (2025). The lab streaming layer for synchronized multimodal recording. *Imaging Neuroscience*, 3, IMAG.a.136. <https://doi.org/10.1162/IMAG.a.136>

LabStreamingLayer. (2025a). *LabRecorder* (1.16.5).
<https://github.com/labstreaminglayer/App-LabRecorder>

LabStreamingLayer. (2025b). *SharpLSL* (1.16.2-rc.1).
<https://github.com/myd7349/SharpLSL>

Lenovo Performance FHD Webcam. (ei pvm.). Lenovo. Noudettu 21. tammikuuta 2026, osoitteesta https://www.lenovo.com/fi/fi/p/accessories-and-software/webcams-and-video/webcams-and-video_webcams/4xc1d66055

Lyhenneluettelo - Kielikello. (2009, huhtikuuta). Kotimaisten kielten keskus.
<https://kielikello.fi/lyhenneluettelo/>

Microsoft. (2022, joulukuuta 10). *What is NuGet and what does it do?*
<https://learn.microsoft.com/en-us/nuget/what-is-nuget>

Millisecond. (ei pvm.-a). *Introducing Inquisit 7*. Noudettu 22. tammikuuta 2026, osoitteesta
<https://www.millisecond.com/support/docs/current/html/introduction.htm>

Millisecond. (ei pvm.-b). *Millisecond*. Noudettu 29. tammikuuta 2026, osoitteesta
<https://www.millisecond.com/>

Millisecond. (2025, joulukuuta 2). *Tower of London Task (ToL)*.
<https://www.millisecond.com/library/toweroflondon>

Open Broadcaster Software | OBS (32.0.4). (2025). <https://obsproject.com/>

Pedersen, M. (2025, tammikuuta 27). *Galvanic Skin Response (GSR) Device Prices - A Comprehensive Guide - iMotions*. iMotions.

<https://imotions.com/blog/insights/galvanic-skin-response-gsr-device-prices-a-comprehensive-guide/>

Polar. (ei pvm.). *Wearing the heart rate sensor*. Noudettu 9. helmikuuta 2026, osoitteesta

https://support.polar.com/en/support/wearing_the_heart_rate_sensor

Polar. (2025a). *Polar H10 | Polar Global*. Polar.

<https://www.polar.com/en/sensors/h10-heart-rate-sensor>

Polar. (2025b). *Polar H10 user manual*.

https://support.polar.com/e_manuals/h10-heart-rate-sensor/polar-h10-user-manual-english/manual.pdf

RealEye. (ei pvm.). *Gaze tracking - Everything you need to know*. Noudettu 5.

helmikuuta 2026, osoitteesta <https://www.realeye.io/features/online-webcam-eyetracking/gaze-tracking>

Shimmer Engineering. (2025). *Shimmer C# API (1.3.10)*. Shimmer Engineering.

<https://github.com/ShimmerEngineering/Shimmer-C-API>

Span, M. (2023). *PolarBand2Is! : Send PolarBand H10 Data to an LSL (1.0.0)*.

<https://github.com/markspan/PolarBand2Is!>

Tobii. (ei pvm.). *Reach further with your research | Tobii Pro Fusion - Tobii*.

Noudettu 22. tammikuuta 2026, osoitteesta

<https://www.tobii.com/products/eye-trackers/screen-based/tobii-pro-fusion>

Turku AMK. (2025, toukokuuta 6). *Asiakaskäyttätymislaboratorio*.

<https://www.turkuamk.fi/palvelu/asiakaskayttaytymislaboratorio/>

Turku AMK. (2026, tammikuuta 2). *INSIGHT Lab - tutkimus- ja kehitysympäristö*

- Turku AMK. <https://www.turkuamk.fi/palvelu/insight-lab/>

Webcam Buying Guide | CDW. (2022, joulukuuta 16). CDW.

<https://www.cdw.com/content/cdw/en/articles/hardware/webcam-buying-guide.html>

xdf-modules. (2026). *pyxdf: Python package for working with XDF files* (1.17.3).
<https://github.com/xdf-modules/pyxdf>

Zhang, C. (2025). *EyeTrax* (0.2.2). <https://doi.org/10.5281/ZENODO.17188537>

Liite 1. Opinnäytetyön lähdekoodit

Kaikki opinnäytetyössä muokatut ohjelmisto on tallennettu GitHub-palveluun.

<https://github.com/Samppa-code/Open-source-solution>

1. Ohjelma A: Shimmer C# Api

- Sijainti: Shimmer3 C# API.cs
- Kuvaus: C# - pohjainen ohjelmakoodi data keräämiseen ja lähettämiseen Shimmer3- laitteella.

2. Ohjelma B: Inquisit 7 Lab

- Sijainti: toweroflondon_inc.iqjs
- Kuvaus: Inquisit-ohjelmointikielellä tehty ohjelma testin suorittamiseen

3. Ohjelma C: PyXDF

- Sijainti: pyxdf.py
- Kuvaus: Koodi lukee annetusta tiedostosijainnista XDF-tiedoston ja piirtää kuvaajat sen tiedoista.