

TARTU ÜLIKOOL  
Arvutiteaduse instituut  
Informaatika õppekava

Harmo-Ron Kallaste

**Tootmisliini kvaliteedijälgimise süsteemi andmebaasi ning töölaua  
loomine**

Bakalaureusetöö (9 EAP)

Juhendaja: Ardi Tampuu, PhD

Tartu 2024

## **Tootmisliini kvaliteedijälgimise süsteemi andmebaasi ning töölaua loomine**

### **Lühikokkuvõte:**

Kvaliteedikontroll on tähtis protsess igas tootmisahelas. Toodangu kvaliteedist ja nõuetele vastavusest aru saamiseks on vaja tulemusi visuaalselt kontrollida. Antud töö kirjeldab, kuidas Nõo Lihatoöstus OÜ tootmisliini tehisintellektipõhisele kvaliteedijälgimise süsteemile arendati lisaks töölaud, andmebaas ning neid ühendav tagasüsteem. See võimaldab kliendil nii reaajas kui ka ajaloolisi andmeid vaadeldes saada kokkuvõtte, milline on liinilt väljuvate pakendite kvaliteet ning mis on peamised probleemid.

### **Võtmesõnad:**

Töölaud, andmebaas, veebirakendus, kvaliteedikontroll, React, Flask, SQL

**CERCS:** P175 Informaatika, süsteemiteooria

## **Dashboard and database development for a production line's quality control system**

### **Abstract:**

Quality control is an important step in each production chain. Visual verification of the results is needed to understand the quality and compliance with requirements of the production. This thesis describes how a dashboard, database, and a backend system were developed for a production line quality control system in Nõo Lihatoöstus, helping the client get an overview of the quality and the common issues of their packaging in real-time and also through historical data.

### **Keywords:**

Dashboard, database, web application, quality control, React, Flask, SQL

**CERCS:** P175 Informatics, systems theory

# Sisukord

Sissejuhatus.....	4
Mõisted ja terminid.....	5
1. Taust.....	6
1.1 Töölauad.....	6
1.1.1 Erinevat tüüpi töölauad.....	6
1.1.2 Interaktiivsus.....	6
1.1.3 Andmete semantika.....	6
1.1.4 Töölaua loomise protsess.....	7
1.1.5 Olemasolevad töölaua loomise platvormid.....	7
1.2 Andmebaasid.....	8
1.2.1 Relatsioonilised andmebaasid.....	8
1.2.2 Mitterelatsioonilised andmebaasid.....	9
1.3 Kvaliteedikontroll.....	11
2. Meetodid.....	13
2.1 Sisendandmete olemus.....	13
2.2 Olemasoleva lahenduse puudused.....	14
2.3 Arendusnõuded.....	15
2.3.1 Planeerimisfaas.....	15
2.3.2 Funktsionaalsed nõuded.....	16
2.3.3 Mittefunktsionaalsed nõuded.....	17
2.4 Tagasüsteem.....	17
2.4.1 Andmebaas.....	17
2.4.2 Tagasüsteem.....	18
2.5 Eessüsteem.....	19
3. Tulemused.....	21
3.1 Lahenduse arhitektuur ja töövoog.....	21
3.2 Andmebaas.....	22
3.3 Töölaud.....	23

3.3.1 Reaalajas jälgimine.....	24
3.3.2 Analüüs.....	26
3.3.3 Ülevaade.....	28
3.4 Teavitused lävenditest kõrvalekalletest.....	30
3.5 Kasutuse stsenaariumid.....	31
3.6 Kliendi tagasiside.....	32
3.7 Potentsiaalsed edasiarendused.....	32
Kokkuvõte.....	34
Viidatud kirjandus.....	35
Litsents.....	36

## Sissejuhatus

Antud bakalaureusetöö eesmärk on arendada andmebaas ning veebipõhine töölaud eksisteerivale kvaliteedikontrolli süsteemile, mis loodi AIRE (ingl *AI & Robotics Estonia*) demoprojektina koostöös Nõo lihatööstusega<sup>1</sup>. See süsteem jälgib tehases kahe kaamera ühte tootmisliini ning kontrollib TI-põhiste lahenduste abil toodete pakendamise kvaliteeti. AIRE projekti lõpu seisuga kuvati tulemusi vaid reaajas, fikseeritud ja kohandamisvõimaluseta jooniste abil ning andmeid ei salvestatud. Käesolev töö lisab suurel hulgal funktsionaalsusi ning lubab teha kokkuvõtlikke ülevaateid erinevate ajaperioodide ning üksikute tootmisliini taktide kohta.

Töölaudade kasutus protsessi kvaliteedi jälgimise eesmärgil on laialt levinud. Kui töölaud on kasutajasõbralikult implementeeritud, on võimalik defektidele ja korduvatele probleemidele kiiremini jälile jõuda [1]. Kvaliteedikontrolli puhul võib töölaud näiteks kasutajale näidata kõiki tuvastatud tulemusi ning vormindada need vastavalt määratud kvaliteedistandarditele. Ühtlasi saab töölaud kõrvalekalletest märku anda – seda näiteks teavituste abil, mis saadetakse probleemi korral kohe töötajatele või teistele vastutavatele isikutele.

Kõiksuguste andmete puhul on vaja neid ka mingil moel hoiustada. Selle jaoks kasutatakse enamasti andmebaase, näiteks PostgreSQL, MySQL, Microsoft SQL Server [2]. Korrektselt üles seatud andmebaas võimaldab kasutajatel luua efektiivseid päringuid ning seadistada süsteem, kus andmete edastamine ning töötlemine ei ole ajaliselt ega arvutuslikult kulukad. Ettevõttes on kasutusel Microsofti poolt arendatud Microsoft SQL Server.

Ettevõtte vajaduste ning soovitud funktsionaalsuste välja selgitamiseks viidi ettevõtte juhiga läbi mitmed koosolekud. Soovitud tulemuste saavutamiseks uuritakse selle töö raames töölaudade loomise tavasid, olemasolevaid raamistikke ja teeke ning võrreldakse erinevaid andmebaaside tehnoloogiaid.

---

<sup>1</sup>[https://github.com/ai-robotics-estonia/2024\\_AI\\_Powered\\_Quality\\_Control\\_Noo\\_Lihatoostus](https://github.com/ai-robotics-estonia/2024_AI_Powered_Quality_Control_Noo_Lihatoostus)

## Mõisted ja terminid

Mõistete ja terminite tõlked ning selgitused põhinevad allikal <https://akit.cyber.ee/>, kui pole teisiti viidatud.

**Töölaud** (ingl *dashboard*) on graafilise kasutajaliidese komponent, mis esitab olulisi koond- või signaalandmeid kompaktselt ja ülevaatlikult, näiteks veebilehena.

**Andmebaas** (ingl *database*) on ettevõtete ja isikute infotöötuse ja -otsingu tarbeks vajalik struktureeritud ja hallatav andmete kogum koos ta halduse tarkvaraga, tavaliselt üks kõige tähtsamaid infovarasid.

**Tagasüsteem** (ingl *back-end*) on kasutajale nähtamatu töötlev, talletav, käitlev jne põhiosa.

**Eessüsteem** (ingl *front-end*) on inimkasutajat või kasutatavat süsteemi tagaosaga liidestav osa.

**API** (ingl *Application Programming Interface*) ehk reeglid ja vahendid rakendusprogrammi suhtluseks operatsioonisüsteemiga, andmebaasihalduse süsteemiga, muu juhtprogrammiga, sideprotokolliga.

**SQL** (ingl *Structured Query Language*) on deklaratiivne programmeerimiskeel relatsiooniliste andmebaasidega suhtluseks.

Järgnevad mõisted on autori kirjeldatud selle töö kontekstis:

**Tootmisliini takt (takt, tootmistakt)** on ajahetk, millal kvaliteedikontrolli süsteem tootmisliinil ühe hulga tuvastusi tegi.

**Tootmistsükkel** on kogumik kõikidest tootmistaktidest, kus toodeti ühte kindlat toodet.

# 1. Taust

Alampeatükid 1.1-1.3 tuginevad Sarikaya et al. artiklil [3].

## 1.1 Töölauad

Töölauade kasutus on üldlevinud ning andmete põhjal otsuste langetamiseks arendatakse ja kasutatakse neid peaaegu igas valdkonnas. Need on muutunud kõik ühes vaadetest interaktiivsete liideste ning mitmete vaadete ja eesmärkidega kujutlusteks. Paljudel juhtudel on implementeeritud erinevad filtrid, mille põhjal kasutaja saab vaadelda temale vajalikke ajavahemikke või tunnuseid.

### 1.1.1 Erinevat tüüpi töölauad

Laialdaselt on töölauad kas visuaalse või funktsionaalse olemusega. Esimesel juhul on need struktureeritud erinevateks komponentideks ehk tegemist on akende paanpaigutusega (ingl *tiled window format*)<sup>2</sup>. Kasutatakse lihtsamaid diagramme ja graafikuid ning suurelt nähtavaid numbreid. Teisel juhul on tegu interaktiivse kuvaga, mis võimaldab reaajas jälgida dünaamiliselt uunevaid andmeid.

### 1.1.2 Interaktiivsus

Interaktiivsuse funktsionaalsus on visualiseerimise puhul ilmne, kuid see võib väga erinevalt realiseeritud olla. Vahel on kasutajale antud võimalus muuta vaadete paigutust või ülesehitust. Mõnikord on töölauad koostatud mitme aknaga kompositsioonina, kus iga aken on mõeldud kindlate komponentide või andmete jaoks. Sageli on võimalik sisse lülitada erinevad filtrid, mille põhjal kuvatakse andmeid kindlate ajavahemike või tunnuste põhjal.

### 1.1.3 Andmete semantika

Töölauade levinud funktsiooniks on ka hoiatuste ja teavituste kuvamine või saatmine, mis aitab tuvastada anomaaliaid. Töötajatele on need suureks eeliseks, kuna nende abil saab kiiresti märgistada probleemseid aspekte ning teha vajalikke parandusi. Sellised töölauad on vahetult ühendatud sisendandmestikuga ning märguanded tekivad kasutaja või ettevõtte poolt määratud lüvendite ja standardite põhjal.

---

<sup>2</sup> <https://akit.cyber.ee>

Kui on defineeritud spetsiifilised lüvendid parameetrite või väärtuste jaoks, siis värvide ja sümbolite kombinatsioonide abil märgendatakse tulemused vastavalt kasutusolevatele standarditele.

Paljud töölaudad on ühenduses andmebaasiga, milles andmed uuenevad regulaarselt. Selliste kasutusjuhtude puhul uuenevad automaatselt ka visualisatsioonid ning muud implementeeritud komponendid. Vastasel juhul on suure tõenäosusega tegemist ajaloolise andmestikuga või käsitletakse konkreetset hetke ajas.

#### 1.1.4 Töölauda loomise protsess

Daryl Orts on kirjutanud artikli [4], mis kirjeldab mitut faasi, mida hea töölauda loomisel silmas peaks pidama. Mitmed soovitusel on võrdväärset tarkvaraarenduse valdkonnas laialdaselt levinud tavadega, kuid on ka selliseid, mis selle töö kontekstis vajavad rohkem tähelepanu:

- **Planeerimise faasis** tuleb võtta aega, et süvenenult tutvuda süsteemiga, kust töölauda andmed tulevad. Eelkõige suuretegevõtete puhul kipuvad andmebaasid ja sealsed tabelid olema väga keerulise ülesehitusega. Mida raskem on vajalikku informatsiooni tagasüsteemist kätte saada, seda rohkem peab olema arendajatel tehnoloogilist pädevust ning seda kauem see protsess aega võtab.
- **Arendusnõuete seadistamise faasis** tuleb välja selgitada, milline peab töölaud olemuselt olema ning sealjuures arvesse võtta mitmete kasutajate personaalseid vajadusi. Iga töölaual oleva komponendi planeerimisel tuleb veenduda, et vajaminevad andmed on kättesaadavad.
- **Disaini faasis** tuleb ajalooliste andmete kuvamise puhul, kui see on planeeritud, kindlaks teha kuidas informatsiooni säilitada ka andmebaasi tõrke korral. Samuti peavad fookuses olema täpsemad päringud ning tuleks defineerida, kui sügavale läbi iga töölaual oleva komponendi analüüsiga minna saab.

#### 1.1.5 Olemasolevad töölauda loomise platvormid

Tänapäeval on enamik töölauda loomise tehnoloogiad seotud otsuseinformaatikaga (ingl *business intelligence*)<sup>3</sup>. Loodud on mitmeid populaarseid tarkvarasid ning platvorme. Katherine Haani artikli [5] sõnul on levinuimad nendest:

---

<sup>3</sup> <https://akit.cyber.ee/>

- **Power BI** ehk Microsofti poolt arendatud kogumik otsuseinformaatika tööriistadest, mis aitab andmeid analüüsida ning leitud tulemusi kergesti teistega näiteks töölaua või raporti kujul jagada [6]. See platvorm on kõige parem valik, kui eksisteeriv tagasüsteem on juba ühenduses Microsofti ökosüsteemiga. Mida suurem on kasutaja organisatsioon ning hajusam süsteem, seda efektiivsemaks nimetatud variant muutub.
- **Tableau**<sup>4</sup> on visuaalse analüütika platvorm, mille eesmärk on võimaldada kasutajal näha andmeid, neid mõista ning nende põhjal tegutseda. See algas 2003. aastal Standfordi ülikooli arvutiteaduse projekti tulemusena, mis proovis parandada analüüsi protsesse ning teha andmed kättesaadavamaks visualisatsioonide abil.

Järgnev lõik on kirjutatud artiklitele [7, 8] tuginedes.

Ise programmeeritud töölaudadega võrreldes on eelnevalt nimetatud tehnoloogiatel eelis loodud lahenduse integreerimisel teiste süsteemidega ning üles seadmise ajakulu ja vajaliku programmeerimisoskuse suhtes. Laialdaselt kasutatud komponendid nagu graafikud, diagrammid ja kohandatavad küljendused on kasutajale valmis tehtud ning vajavad ainult peenhäälestust. Töölaudade või analüüside loomine ei nõua ka põhjalikke tehnoloogilisi teadmisi ning kohati on need ebavajalikud. Küll aga, mida spetsiifilisemad on firma või kasutaja nõuded ning vajadused, seda suurem on tõenäosus, et neid pole võimalik mainitud platvormide siseselt teostada. Seda seetõttu, et kogu loogika on piiratud tarkvara pakkuja loodud süsteemi ja selle võimaluste poolt.

## 1.2 Andmebaasid

See alampeatükk põhineb Wajid Ali et al. kirjutatud artiklil [9], kui teisiti viidatud pole.

Asjatundjate sõnul kahekordistub informatsiooni hulk maailmas igal aastal, seega selle efektiivne hoiustamine on aina keerulisem. Parim lahendus probleemile on andmebaaside kasutus. Need võimaldavad andmete struktureeritud salvestamist, lugemist ning otsimist ja kätte saamist. Suures pildis jagunevad andmebaasid kaheks - relatsioonilised ning mitterelatsioonilised.

### 1.2.1 Relatsioonilised andmebaasid

Relatsioonilistes ehk SQL andmebaasides hoitakse andmeid tabelites, mille vahel on võimalik luua relatsioone. Ühtlasi on selliste andmebaaside individuaalsed read oma

---

<sup>4</sup> <https://www.tableau.com/why-tableau/what-is-tableau>

olemuselt relatsioonilised ehk tabeli tulbad on omavahel seotud [10]. Joonisel 1 on näiteks üks tabel töötajate ja teine osakondade kohta. Mõlema puhul hoiame seal ainult seda informatsiooni, mis on seotud selle kindla olemiga. Esimeses tabelis võib hoiustada näiteks töötaja nime, ametinimetust, palganumbrit ning viidet sellele osakonnale, kus ta töötab. Selle viite abil saab leida kindla siisekande teisest tabelist, kus saab hoiustada näiteks osakonna nimetust, kontori asukohta ja muud.

**Employee table**

Empno (PK)	Ename	job	Mgr	hiredate	sal	comm	Deptno (FK)
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	BLAKE	MANAGER	7839	01-MAY-81	2850		30

**Department table**

Deptno (PK)	dname	loc
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO



**Joonis 1.** Relatsioonilise andmebaasi tabelid töötajate ja osakondade kohta<sup>5</sup>

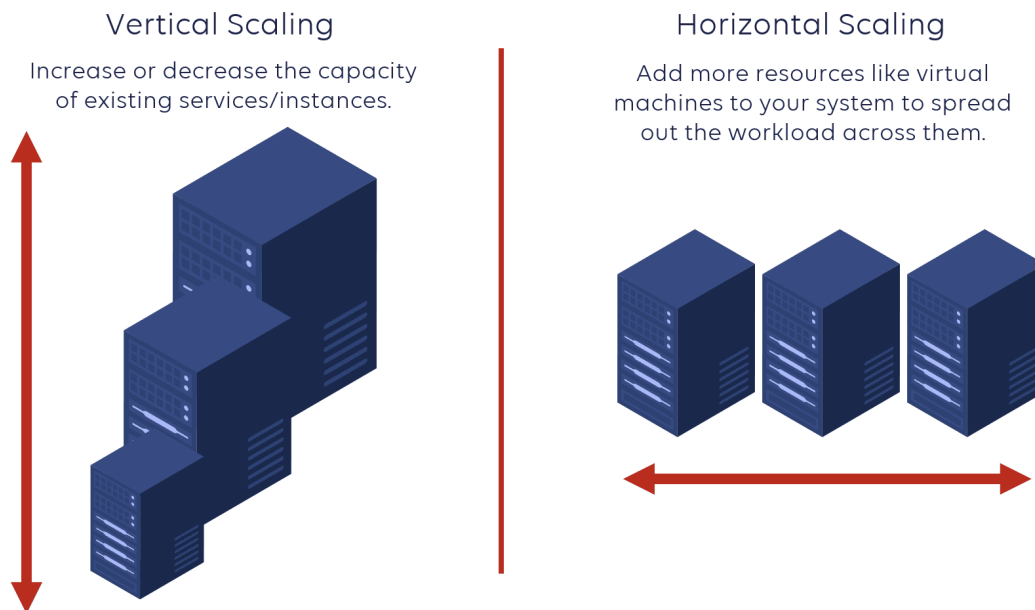
Andmemahud on tehnoloogia arenguga plahvatuslikult kasvanud ning andmed pole enam alati piisavalt kindla ülesehitusega. Relatsioonilised andmebaasid skaleeruvad vertikaalselt, ehk nende jõudlust saab suurendada olemasoleva serveri komponentide parandamisega, kuid mitte serveri hajutamisel (vt. Joonis 2). 20. sajandi lõpus tekkis uus kontsept NoSQL, mis on mõeldud selliste juhtude korral kasutamiseks.

### 1.2.2 Mitterelatsioonilised andmebaasid

Mitterelatsiooniliste ehk NoSQL andmebaaside põhimõte seisneb kindla struktuurita ülesehituses ning horisontaalses skaleeritavuses. Viimane tähendab, et erinevalt relatsioonilisest andmebaasist ei pea NoSQL andmebaasi jõudluse suurendamiseks serverile ehk arvutile rohkem mälu või rohkemate tuumadega protsessorit paigaldama, vaid

<sup>5</sup> <https://www.datacamp.com/tutorial/sql-reporting-analysis>

andmebaasi saab üle mitme serveri laiai hajutada (vt. Joonis 2). NoSQL andmebaasid on suurepärased suurandmete ning kiiresti tekkiva informatsiooni hoiustamiseks.



**Joonis 2.** Serveri horisontaalne vs. vertikaalne skaleerimine<sup>6</sup>

Gaurav Vaish on kirjutanud raamatu [11], kus on välja toodud mõned erinevad NoSQL andmebaasi tüübid:

- **Column-oriented databases** - Kui relatsioonilistes andmebaasides kujutatakse andmeid kahedimensionaalsetes tabelites ning hoitakse, päritakse ja töödeldakse ridadena, siis selle lähenemise puhul hoitakse andmeid tulpadena. Selliste andmebaaside puhul on enamasti võimalik uusi tulpasid lisada, muretsemata selle pärast, kuidas varasemad andmed uue tulba suhtes ühtlustada.
- **Document store** - Sellised andmebaasid võimaldavad lisada, pärida ja muuta poolstruktureeritud andmeid. Enamasti kasutavad antud lahendused XML, JSON, BSON või YAML failiformaate ning suhtlus toimub HTTP protokolliga ja RESTful API abil. Nimetatud andmebaaside kasutamise eeliseks on see, et puudub kindel skeem, seega andmed võivad olla erinevat tüüpi ning nende sisu ja olemus võib aja jooksul dünaamiliselt muutuda.

<sup>6</sup> <https://www.stormit.cloud/blog/scalability-in-cloud-computing-horizontal-vs-vertical-scaling/>

- **Key-value store** - Tegemist on sarnase ideega nagu *document store*, aga andmete loomisel, muutmisel ja pärimisel tuleb ise võti täpsustada. Enamjaolt on sellised andmebaasid kasutusel vahemäluna, kuid need võimaldavad ka püsivat salvestamist.
- **Graph store** - Sellises andmebaasis on andmete vahelised seosed ja ühendused kujutatud graafidena. Näiteks on see hea valik sotsiaalse võrgustiku või serverite vaheliste ühenduste esitamisel. Neid on mõistlik kasutada juhul, kui andmete vahel on tugevad seosed ehk relatsioonid, kuid sellegipoolest on soov kasutada NoSQL andmebaasi eeliseid relatsioonilise andmebaasi asemel. Näiteks saab hoida andmeid *document store* tüüpi andmebaasis ja nende vahelisi suhteid *graph store* andmebaasis.

### 1.3 Kvaliteedikontroll

Järgnev alapeatükk baseerub Amitava Mitra kirjutatud raamatul [12].

Kvaliteedikontroll on üldistatult süsteem, mis proovib toote- või teenusepõhise tagasiside läbi hoida spetsiifilist standardit ning lahknevuste korral kasutada parandusmeetmeid. Selle võib grupeerida kolmeks kategooriaks:

- *Off-Line Quality Control* - Toodangu ja tootmisprotsessi kavandamine nii, et toodangu erinevus standardist oleks minimaalne. Võimalikult palju tehakse enne tootmisprotsessi ära. Jaapani inseneri Genichi Taguchi ideede kohaselt on selle peamiseks eesmärgiks korraldada tootmine võimalikult mürakindlalt, et protsess oleks valguse, tolmu, niiskuse ja teiste ettearvamatuste suhtes vähetundlik.
- *Statistical Process Control* - Protsessi väljundit või tulemust võrreldakse standardiga ning kõrvalekalde korral tehakse parandusi. Näiteks kontrollitakse päeva jooksul koostatud paberdokumentide kvaliteeti nii, et õhtu lõpus kogutakse informatsioon kokku ning erisuste korral proovitakse mõista, kuidas need tekkida võisid. Vastavate mõjutegurite suhtes viiakse läbi parandused ehk näiteks viiakse läbi lisakoolitused töötajatele või lihtsustatakse tööprotsessi.
- *Acceptance Sampling Plans* - Määratakse kindel uuritava partii suurus, aksepteeritav ebakvaliteetse toodangu kogus ning vastuvõetavuse tingimused ja uuritakse nende põhjal toodangut või teenuse tulemusi. Oletame, et kaupa toodetakse 500 partiides, valitud uuritava partii suurus on 50, aksepteeritav ebakvaliteetse toodangu kogus on 3 ning toote kvaliteetne paksus on  $12 \pm 0.4$  mm. Plaani kohaselt valitakse 500 tootest

suvalised 50 ning kui nende seast 3 või enam on ebakvaliteetsed, siis terve partii ehk kõik 500 toodet loetakse praaktoodanguks.

Selle lõputöö sisendandmete puhul on kasutusel *statistical process control* ehk võrreldakse protsessi käigus saadud väljundit lävendiga, antakse märku, kui erinevus on ilmne ning proovitakse kindlaks teha, kas protsess suudab toota piisavalt korrektse tulemuse. Täpsemalt võib olukorra kohta öelda *online statistical process control*, kuna informatsiooni kogutakse paralleelselt toote, protsessi või teenuse funktsioneerimise ajal. Probleemi korral tehakse koheselt parandused ehk süsteem proovitakse aksepteeritavasse staadiumisse viia võimalikult kiiresti. Sellisel moel minimeeritakse praaktoodangu mahtu või aega, kus eeldatav protsess ei tööta nii nagu peaks.

## 2. Meetodid

Alampeatükid 2.1 ja 2.2 põhinevad allikatel [13, 14].

### 2.1 Sisendandmete olemus

Möödunud aasta jooksul teostasid Tartu Ülikooli arvutiteaduse instituut ning Nõo lihatööstus demoprojekti, mida rahastas AIRE. Projekti eesmärgiks oli välja uurida kas tehisintellekti, täpsemalt masinnägemise, abil on võimalik mitmeid kvaliteeti mõjutavaid aspekte samaaegselt kontrollida. Tooteliini kohale paigaldati kaks kaamerat (vt. Joonis 3, 4) ning töödeldavad andmeid tulid RTSP (ingl *Real Time Streaming Protocol*) kaudu tagasüsteemi.



**Joonis 3.** Vasaku kaamera jäädvustus



**Joonis 4.** Parema kaamera jäädvustus

Toodete puhul jälgiti järgnevaid tunnuseid:

- säilimisaja korrektsus/olemasolu.;
- siltide korrektsus/olemasolu pakendil;
- pakendis oleva toote olemasolu/vastavus siltidele.

Igas tootmistaktis satub kaamerate alla 8 või 12 toodet. Igal taktil leiti enim tuvastatud triipkood ning päriti selle kohaselt toodete andmestikust, mis on oodatav kuupäev, siltide ning toote välimus. Andmestik seadistati Nõo lihatööstuse ning projekti arendajate koostöös. Lisaks oli lahenduse ülesandeks ka ära tunda hetk, millal tooteliin liikus ning uued tooted

kaameravaatesse tulid. Antud töö autor kuulus selle projekti arendusmeeskonda Tartu Ülikooli poolel.

## 2.2 Olemasoleva lahenduse puudused

Kuigi see polnud demoprojekti arendusplaanis prioriteediks, loodi tuvastuste tulemuste kuvamise jaoks käsurea põhine visualiseerimise loogika ning veebipõhine töölaud.

Käsureal kuvamine (vt. Joonis 5) oli kasulik ainult reaajas jälgimiseks, kuna seal olid näha peamised tulemused, kuid need ei olnud kuidagi seotud varasemate andmetega ega keskmistatud. See lihtsustas arendustiimi tööd, kuid kliendi perspektiivis oli keeruline aru saada, kust probleemid või ebakorrektsed tulemused tekkida võisid – tagantjärei polnud võimalik näha andmeid (RTSP striimi kuvatõmmiseid), mille põhjal tagasüsteem töötas.

```
Decoded 4 barcodes.
Decoded 1 barcodes.
[ 'Prosciutto Cotto 105 g', 'Prosciutto Cotto 105 g', 'Prosciutto Cotto 105 g', 'Noo rulaad 135 g', 'Noo rulaad 135 g', 'Noo rulaad 135 g', 'Noo rulaad 135 g', 'Noo rulaad 135 g' ]
===== FRAME RESULTS 1 =====
The most detected barcode in this frame was: 4740574008052, Noo rulaad 135 g
The product name is: Noo rulaad 135 g
The most detected product during validation in this frame was: 4740574008052, Noo rulaad 135 g
The detected dates in this frame were: ['24.12.2024', '24.12.2024', '24.12.2024', '24.12.2024', '24.12.2024', '24.02.2024']
The expected expiry date was: 24.12.2024 62.5%

Label detection results
-----
| True | True | True | True |
| True | True | True | True |
| True | True | True | True |
| True | True | True | True |
-----

Data successfully sent to the dashboard at http://127.0.0.1:5000/apl/receive-data
total time 2.2457275390625
```

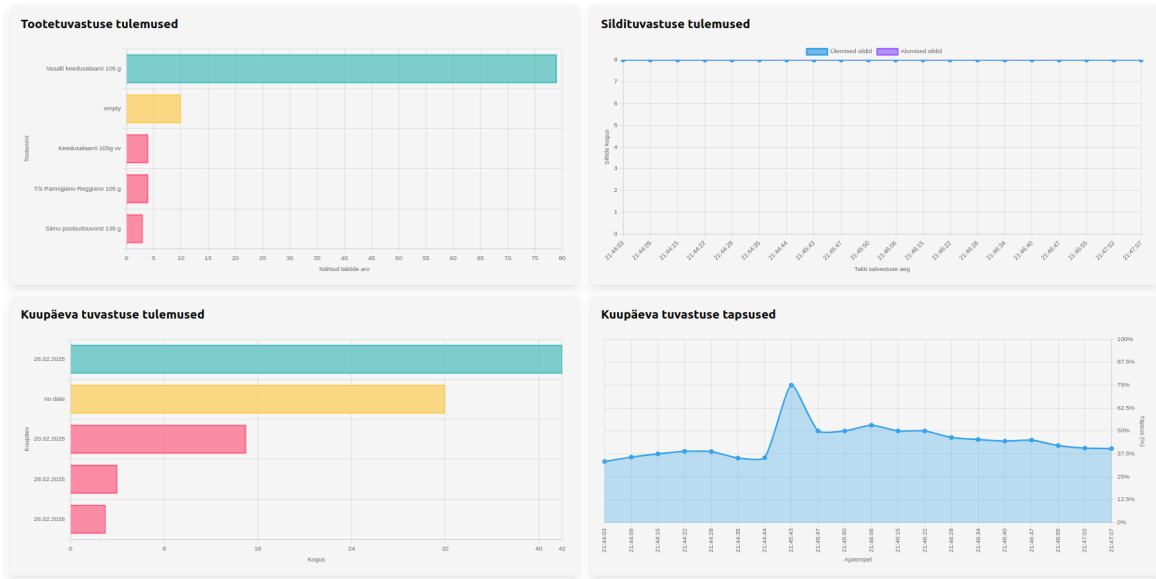
Joonis 5. AIRE projekti raames valminud käsurea põhine visualiseerimine<sup>7</sup>

Projekti raames loodud töölaud (vt. Joonis 6) andis ülevaate viimase 20 tootmistakti jooksul tehtud tuvastustest. Joograafikute ja tulpdiaagrammide näol olid kujutatud järgmised tulemused:

- tuvastatud siltide arv iga tootmistakti puhul (joondiagramm);
- eeldatud (õige) kuupäeva osakaal tuvastuste hulgas (joondiagramm);
- tuvastatud kuupäevade jaotus (tulpdiaagramm, kokkuvõtvalt viimase 20 takti andmetel);
- tuvastatud toodete jaotus (tulpdiaagramm, kokkuvõtvalt viimase 20 takti andmetel).

<sup>7</sup>[https://github.com/ai-robotics-estonia/2024\\_AI\\_Powered\\_Quality\\_Control\\_Noo\\_Lihatoostus](https://github.com/ai-robotics-estonia/2024_AI_Powered_Quality_Control_Noo_Lihatoostus)

Praegune toode liinil on: **4740574009820 --- Vasalli keedusalaami 105 g --- 26.02.2025**



**Joonis 6.** AIRE projekti raames valminud töölaud<sup>8</sup>

Kliendi hinnangul oli ka selle lähenemise puhul mitmeid puudujääke. Näiteks ei olnud võimalik kasutajal muuta vaadeldavate taktide arvu, kuna see oli püsiprogrammeeritud viimase 20 takti peale. Samuti ei olnud see reaajas jälgimisel väga mugav, kuna iga diagrammi puhul pidi spetsiifilisi punkte jälgima ehk puudus ülevaade.

## 2.3 Arendusnõuded

Nõo lihatööstuse tegevjuhi ning antud töö autori vahel toimus ajavahemikul veebruar-mai vestlus, et välja selgitada kasutajate pooled vajadused ning nõuded loodavale lahendusele. Samuti tutvustati arendusetappe järk-järgult, et klient saaks lähenemistes veenduda. Suhtlus toimus meili ja videokõnede vahendusel. Järgnevalt võtab autor kokku selle suhtluse tulemusena välja selgitatud nõuded.

### 2.3.1 Planeerimisfaas

Algselt tegi klient ülevaate tema poolt pakutavatest ressurssidest ning kontrolliti, et autoril oleks kõikidele vajalikele seadmetele ja süsteemidele ligipääs. Uuriti, mis kujul olemasolev kvaliteedikontrolli süsteem tulemusi väljastab ning kas on vaja ja kui keeruline oleks seda formaati muuta. Mõeldi ka töölaua ja andmebaasi võimalike funktsionaalsuste peale, mida

<sup>8</sup>[https://github.com/ai-robotics-estonia/2024\\_AI\\_Powered\\_Quality\\_Control\\_Noo\\_Lihatoostus?tab=readme-ov-file](https://github.com/ai-robotics-estonia/2024_AI_Powered_Quality_Control_Noo_Lihatoostus?tab=readme-ov-file)

tulevikus selle töö väliselt lisada saab, et arhitektuur või kindel süsteemi osa saaks seadistatud nii, et edasiarendus oleks lihtsam.

### 2.3.2 Funktsionaalsed nõuded

Siin on välja toodud nimekiri käesoleva töö funktsionaalsetest nõuetest:

- Töölaud kui eessüsteem peab olema veebipõhine ning see peab olema kas serveris või loodud nii, et see oleks sinna kergesti tõstetav.
- Töölaua peab olema reaalajavaade, mis kuvab viimase tuvastatud takti kohta järgneva informatsiooni:
  - Jäädvustused kaamerastriimist, mille põhjal kvaliteedikontrolli tagasüsteem tuvastusi ja arvutusi tegi.
  - Teave enim tuvastatud triipkoodile vastava toote kohta.
  - Tulpdiagrammid kuupäeva ja toote tuvastuse tulemuste kohta, kus on välja toodud 5 enim tuvastatud kuupäeva/toodet ning on selgelt näha, kui suure osa kõikidest tuvastustest moodustasid oodatud tulemused.
  - Tulpdiagramm, mis näitab sildituvastuse tulemusi ning suudab eristada kahe kaamera ja kahe sildiga pakendite puhul ülemiste/alumiste siltide vahel.
  - Ülevaatliskujutus kõikidest tuvastustest antud tootmistaktis, kus värvide ja kujundite põhjal on aru saada, mis oli toodete puhul õigesti ja mis valesti.
- Töölaua peab olema analüüsi vaade, mis võimaldab:
  - Valida ühe kindla toote kõikide seni nähtud toodete hulgast (vaikeväärtus peaks olema sellel hetkel tootmisliinil olev toode).
  - Valida kindel taktide arv  $N$ , mille maksimum on 100 (vaikeväärtus on 100).
  - Näha sildituvastuse tulemusi joondiagrammi näol, kus on iga viimase  $N$  takti kohta näha mitu ülemist ning kahe sildiga pakendite puhul ka alumist silti tuvastati (ühe sildiga toodete puhul loeb kvaliteedikontrolli süsteem seda ülemise sildina).
  - Näha sildituvastuse tulemust, eeldatud välimusega siltide osakaalu, kolme joonega joondiagrammi näol, kus vasaku ning parema kaamera kohta on vastavalt üks joon ning kolmas joon näitab kahe kaamera tulemuste summat.
  - Näha toote ja kuupäeva tuvastuse tulemusi samal moel nagu reaalajavaates, aga andmehulgaks on ühe tootmistakti asemel  $N$  takti.

- Toote ja kuupäeva tuvastuse tulemusi, eeldatud välimuse/kuupäeva määra, joondiagrammidena, kus mõlemal juhul on arvutatud:
  - eeldatud kuupäevade/toodete osa kõikidest kuupäevadest/toodetest;
  - puuduvate kuupäevade/toodete osa kõikidest kuupäevadest/toodetest.
- Töölaud peab jälgima erinevate tuvastuste jaoks kindlalt määratud lävendeid ning nende ületamise korral salvestama teavituse juhtunu kohta, tehes viite ajavahemikule, mille jooksul ületamine toimus.
- Tagasüsteem peab olema kas serveris või loodud nii, et see oleks sinna kergesti tõstetav.
- Kogu süsteem peab lõpuks töötama ettevõtte sisevõrgus.

### 2.3.3 Mittefunktsionaalsed nõuded

Siin on välja toodud nimekiri käesoleva töö funktsionaalsetest nõuetest:

- Kõikidel graafikutel peavad olema pealkirjad ja/või legendid, et uuel kasutajal oleks neid võimalikult kerge mõista.
- Võimalusel tuleb kasutada värve:
  - roheline – hea/korrektne/oodatud;
  - punane – halb/ebakorrektne/mitteoodatud;
  - kollane – puudub.
- Kõik graafikud, mis ei väljenda otseselt tõeväärtuslikke tulemusi, võiksid võimalusel kasutada Nõo lihatööstuse värve.

## 2.4 Tagasüsteem

### 2.4.1 Andmebaas

Kuna salvestatavate andmete struktuur on reeglipärane ning on seos tootmisliini taktide ja vastavate tulemuste vahel, siis osutus valituks relatsiooniline SQL andmebaas. Kliendil on andmebaasina kasutusel Microsoft SQL Server, mis on Microsofti poolt arendatud relatsiooniliste andmebaaside jaoks mõeldud haldussüsteem [15]. Kerge ühilduvuse ja sama tehnoloogiaga jätkamise eesmärgil oli see kasutusel ka antud töös. Arenduse kiiruse ja lihtsuse tagamiseks oli protsessi käigus kasutusel Microsofti ametlik Dockeri pilt, et arendustöö oleks lokaalne ja kergesti hallatav. Docker on avatud lähtekoodiga platvorm, mis pakendab rakenduse ja kõik selle jaoks vajalikud sõltuvused konteinerisse, töötades

operatsioonisüsteemi kerneli peal isoleeritud moel [16]. Lõplik andmebaas seadistati ettevõtte serveris olevasse virtuaalmasinasse eelnevalt väljatoodud haldussüsteemi abil.

#### 2.4.2 Tagasüsteem

Varasemas, AIRE projekti raames loodud lahenduses oli tagasüsteemina Flask, mille abil vahendati andmeid eessüsteemi. See on Pythoni programmeerimiskeelel põhinev mikroraamistik, mis võimaldab kiiret veebirakenduste loomist [17]. Kuna tegemist on võimeka ja levinud tehnoloogiaga, oli kasutusel see ka antud töö raames. Flask ise ei paku andmebaasiga suhtlust võimaldavat liidest, seega selleks kasutati teeki pyodbc<sup>9</sup>. Tagasüsteemi peamisteks ülesanneteks olid suhtlus andmebaasi ja vahemäluga, andmehulkadest omakorda osahulkade koostamine ning tuvastuste täpsuste ja muude arvutamiste läbiviimine. Näiteks tuleb iga tootmisliini takti puhul AIRE tagasüsteemist kaks listi toote- või kuupäevatu vastuse tulemustega – üks näitab kas antud tuvastus puudus, oli vastavuses ootusega või mitte, ning teine näitab, mis toode või kuupäev tuvastati. Selle andmestiku põhjal koostab selle töö tagasüsteem kogumiku, mis näitab viit enim tuvastatut toodet või kuupäeva ning kui suure osa seda tüüpi tuvastus moodustas kõikidest tuvastustest.

Kuna andmeid tuli tagasüsteemi tihti, umbes iga 7 sekundit kestva takti järel, siis pidevalt päringute tegemine andmebaasi oleks olnud liialt kulukas. Seetõttu implementeeriti vahemälu, milleks kasutati Redis platvormi. Tegemist on äärmiselt kiire residentse mitterelatsioonilise kaugandmebaasiga, kus võtmetele vastavad väärtused, mis võivad olla viit erinevat tüüpi – sõne, järjend, hulk, räsi ja sorteeritud hulk [18]. See võimaldas hoiustada andmeid ning nende põhjal genereeritud osahulkasid, mis ei muutu liiga tihti või on pidevas kasutuses. Lähenemine garanteeris ka selle, et igal muul juhul, välja arvatud siis kui terve süsteem algselt tööle pandi, oli eessüsteemile kohe võimalik vajalikud andmed edastada. Tagasüsteemi probleemi või andmebaasiga ühenduse katkemise korral kõik salvestatu säilis, kuna vahemälu jooksis eraldi Docker konteineris. Järgnevalt on välja toodud mõned vahemälu hoitavad andmed ning põhjendatud, miks see kasulik on:

- Kogumik kõikidest erinevatest toodetest, mis andmebaasis saadaval on. Sellela oleks pidanud iga uue tootmisliini takti korral kas tegema päringu andmebaasile või hoidma seda Python'i muutujas. Esimese korral peaks andmebaas kasutama spetsiaalseid vaateid, et see operatsioon poleks arvutuslikult liiga kulukas. Teisel juhul ei säiliks see informatsioon erinevate käitusaegade vahel.

---

<sup>9</sup> <https://pypi.org/project/pyodbc/>

- Üldine informatsioon antud hetkel tootmisliinil oleva toote kohta - nimi, EAN13 kood, paigutustüüp tooteliinil (kas 8 või 12 toodet) ning ajahetk, millal hakkas selle toote viimane nähtud tootmistsükkel. Andmed ei muutu tihti, kuna tootmine käib toote põhistes tsüklites. Vahemälu kasulikkuse põhjendus on analoogne eelmise punktiga.
- Iga tuvastuskategooria täpsused ja osahulgad iga toote viimase 100 tootmistakti kohta. See oli vajalik, kuna selle informatsiooni põhjal kontrolliti jooksvalt kvaliteedi lävendeid, iga uue tootmistsükli korral oli kiiresti võimalik näha antud toote varasemaid tootmistakte ning nende jooksul tehtud tuvastuste tulemusi ja täpsusi. Andmebaasi skeemi kohaselt neid jäädavalt ei salvestata, vaid arvutatakse kas ümber või uuesti iga uue tootmistakti korral, seega antud kontekstis mälu suhtlus on vahemälu kasutamisel kiirem.

Iga tootmistakti korral saadab kvaliteedikontrolli süsteem ka mõlema kaamera jäädvustused, mida on vaja salvestada. Piltide hoiustamine SQL andmebaasis on võimalik, aga see pole soovitatud. Kui vaadata näiteks töö kirjutamise ajal pilvetehnoloogia valdkonna suurimaid ettevõtteid nagu näiteks AWS (ingl *Amazon Web Services*) jt., siis piltide ja laiemalt meediaobjektide jaoks on spetsiaalsed teenused ja virtuaalmasinad. Selle töö käigus kasutati platvormi nimega Cloudinary, mis on meedia töötlemise, hoiustamise ja vahendamise jaoks mõeldud pilveplatvorm [19].

## 2.5 Eessüsteem

Töölaud pidi olema nii visuaalne kui ka funktsionaalne. Teada olid paljud standardid ja lävendid, seega andmete semantilisuse koha pealt tuli kasutusele võtta värvid ja varieeruvad suurused, et väljendada erinevuseid normidest. Tootmise puhul on olulised ka kõikide, nii defektiga kui ka korrektsete toodete arvud ja liikuvad keskmised, mis tähendab, et kasutusel pidid olema graafikud või diagrammid. Kuna kasutaja peab saama toodangut jälgida ja analüüsida nii reaajas kui ka retrospektiivselt, siis töölaua kompositsioon pidi olema mitme vaatega, et oleks võimalik fikseerida kindel huvipunkt.

Kuna töö käsitusala ei olnud liiga lai ning andmed on seotud üsnagi ebastandardsete olemitega (lihatoodete pakenditega), mitte äriprotsesside, kliendisuhete või näiteks sotsiaalmeedia statistikaga, siis ei kasutatud eksisteerivat töölaua loomise tarkvara nagu Power BI või Tableau.

Järgnevas lõigus on meetodite kirjeldused kirjutatud Khoi Phani bakalaureusetöö [20] põhjal.

Töölaud ise on olemuselt SPA (ingl *single-page application*). Tegemist on veebiarenduse meetodiga, kus kasutajale kuvatav sisu uueneb dünaamiliselt ühe veebilehekülje raames. Vajalikud andmed tulevad tagasüsteemist ehk serverist ning esitusvalmendus (ingl *rendering*)<sup>10</sup> toimub kasutaja ehk kliendi poolel, seega navigatsioon veebilehe erinevate vaadete ning osade vahel on kiirete ja reaajas toimuvate uuenduste tõttu sujuvam. Selle töö arendusnõutete põhjal pidi töölaud olema mitmevaateline, seega veebirakendus oleks saanud olla ka MPA (ingl *multi-page application*), kus iga kasutajasisendi korral saadab server uuendatud staatilise veebilehe kasutajale. Selle meetodi tunneb ära, kui sama veebirakenduse siseselt muutub kasutajasisendi põhjal näiteks URL. Sedaviisi üles seatud eessüsteemid on otsingumootoritele kättesaadavamad, kuna iga lehekülge on võimalik kergesti indekseerida. Küll aga ei ole MPA omadustest selle töö puhul kasu, sest lõpplahendus on kasutatav ainult kliendi poolt ning kasutajakogemus ja veebilehe sujuvus on prioriteetid.

Eessüsteemi arendamiseks kasutati teeki React, mida programmeeriti Typescript keeles. React on kliendipoolne JavaScript'i teek, mis sai alguse 2011. aastal Facebooki vajadusest veebirakenduse kasutajaliideses muutuseid efektiivselt hallata ja optimeerida [21]. See võimaldas kergesti süsteemi korduvkasutatavateks koodijuppideks jagada ning efektiivselt parameetreid ja sisendatribuute (ingl *props*) nende vahel edastada. Kuigi veebiarenduse valdkonnas on professionaalide hulgas populaarseteks alternatiivideks ka Angular<sup>11</sup>, Vue.js<sup>12</sup> ja paljud teised, siis autori varasema kogemuse ning React'i sisseehitatud funktsionaalsuste tõttu sai see valituks [22]. Samuti on tegemist teegiga, mille omadused on eriti sobilikud just SPA-de puhul. Seda just modulaarse komponendi põhise tööpõhimõtte ja VDOM-i (ingl *virtual domain object model*) tõttu ehk kasutajasisendi põhjal saab süsteem kiiresti aru, millised on veebilehel uuendamist vajavad osad ning ülejäänud jätab puutumata.

Komponendid said küljendatud toetudes shadcn<sup>13</sup> teegile, sest see pakkus modernse ja minimalistliku disainiga elemente, mida sai soovi korral veel ise kas jäädavalt või ajutiselt muuta. Lisaks tegi see ka arendusprotsessi kergeks, kuna valikus olevaid elemente sai käsurea abil projekti soovikohaselt projekti lisada. Algselt oli plaanis kasutada teeki MaterialUI<sup>14</sup>, kuid see ei võimaldanud soovitud funktsionaalsuste implementeerimist.

---

<sup>10</sup> <https://akit.cyber.ee>

<sup>11</sup> <https://angular.dev/>

<sup>12</sup> <https://vuejs.org/>

<sup>13</sup> <https://ui.shadcn.com/>

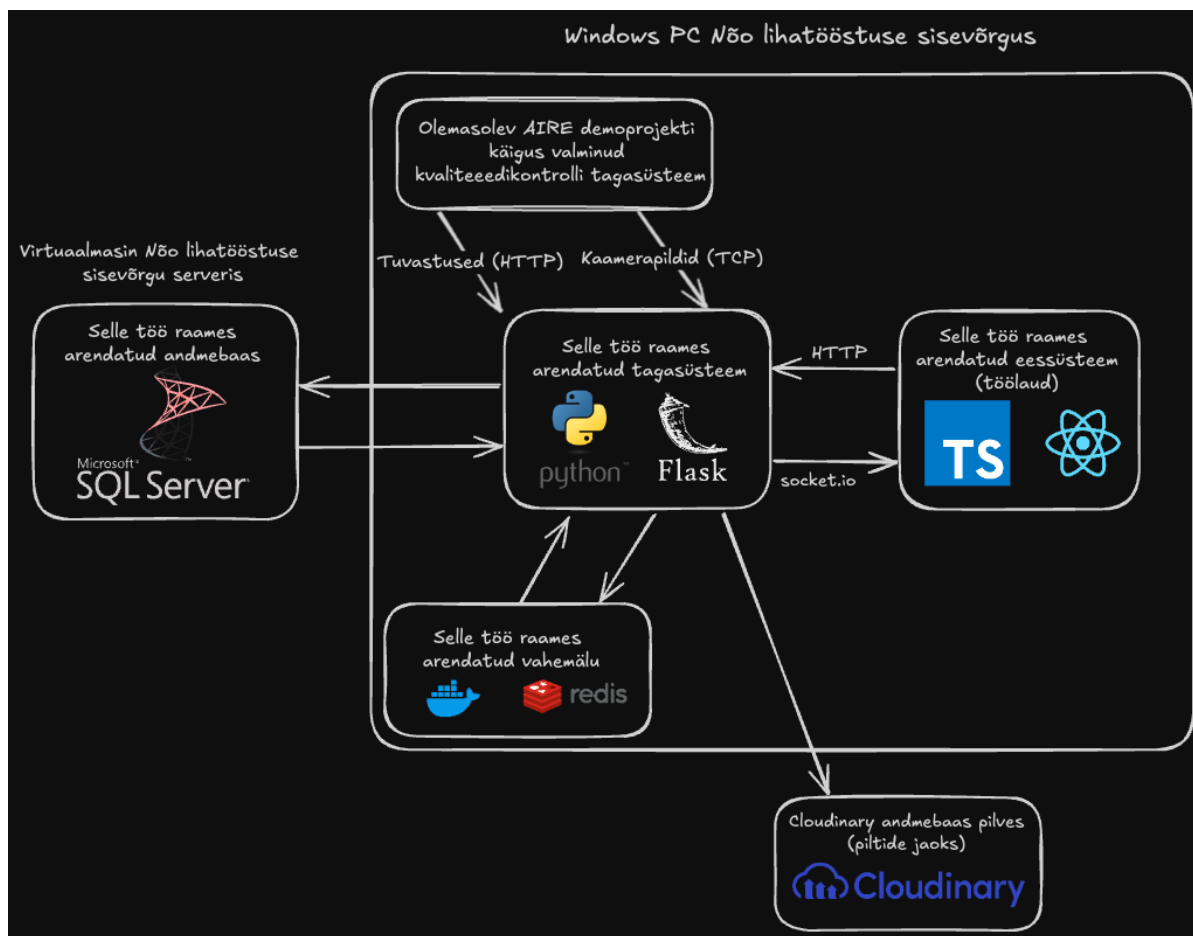
<sup>14</sup> <https://mui.com/>

### 3. Tulemused

Koostööpartneri soovil ei ole käesoleva lõputöö raames valminud programmikood avalikult kättesaadav. Kood on talletatud privaatses repositooriumis ning sellele ligipääsu saamiseks tuleb pöörduda autori poole e-posti teel: *harmoronkallaste.thesis@gmail.com*.

#### 3.1 Lahenduse arhitektuur ja töövoog

Järgneval joonisel on näha lahenduse arhitektuur skeemina.



Joonis 7. Lõpplahenduse arhitektuur

Iga uue tootmisliini takti puhul teeb AIRE demoprojekti raames loodud tagasüsteem tuvastused ja pildid, mille põhjal kvaliteedikontroll toimus, ning edastab need vastavalt HTTP ja TCP päringute abil selle töö jaoks arendatud tagasüsteemi. Seal töödeldakse need kõigepealt andmebaasi skeemile vastavale kujule. Seejärel need sisestatakse andmebaasi ja selle operatsiooni õnnestumise korral tehakse päring ning saadakse see sama informatsioon tagasi. See on vajalik, sest töölauale ei tohi jõuda andmeid, mida hiljem andmebaasist kätte

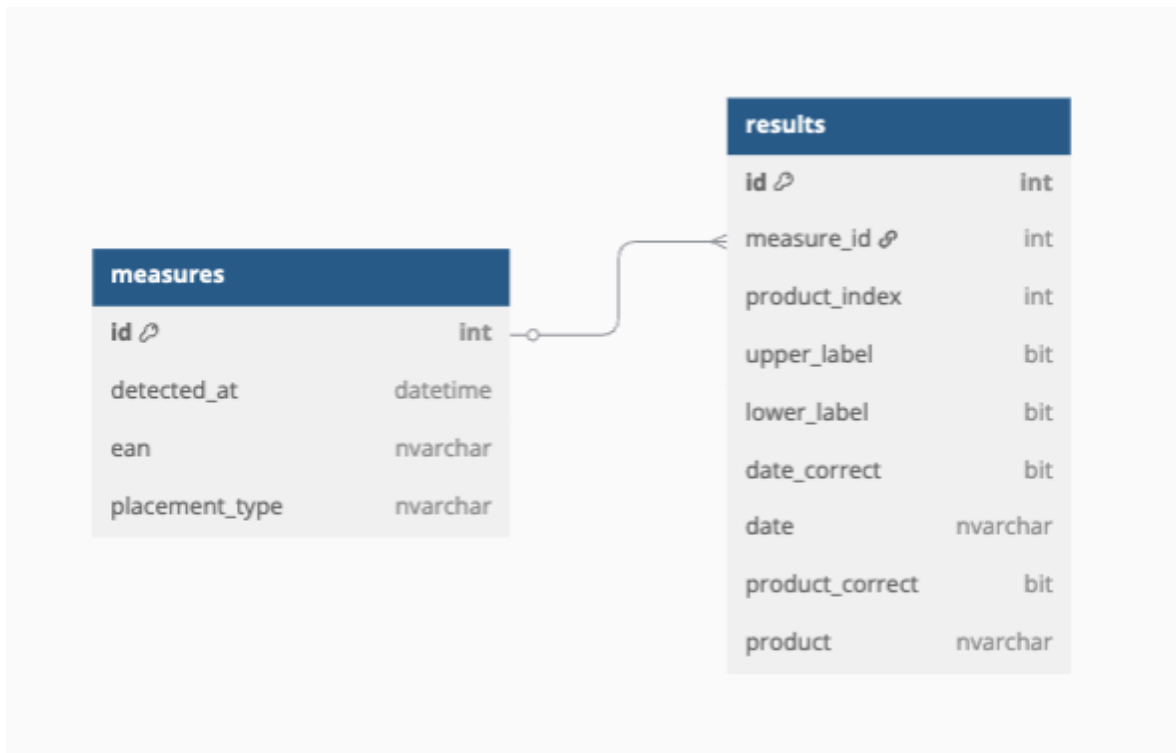
pole võimalik saada. Selle protsessi järel on teada ka äsja lisatud tootmistakti identifikaator, mida kasutatakse saabunud piltide salvestamisel Cloudinary pilve. Kuna eessüsteemi erinevad komponendid on mõeldud kindlate tuvastustulemuste visualiseerimiseks, siis järgmisena toimub erinevate osahulkade genereerimine. Näiteks arvutatakse sildituvastuse täpsused ning grupeeritakse tuvastused vastavalt toodete asukohtadele liinil, sest kvaliteediprobleem võib ilmneda just ühe kaamera all või kindlas toodete tulbas/reas. Seejärel uuendatakse vahemälus olevaid väärtusi ning tehakse andmebaasile päring, mis tagastab sellel hetkel tooteliinil oleva toote viimased 100 kvaliteedikontrollist läbi käinud takti ning nende jooksul tehtud tuvastused. Viimasena kontrollitakse, kas toote viimase 50 tootmistakti põhjal on mõni tuvastuskategooria tulemus ehk täpsus alla seadistatud lävendi, millisel juhul koostatakse teavitus, või mitte. Kui kõik vajalikud osahulgad on genereeritud, jäädvustused pilve salvestatud, vahemälu uuendatud ja lävendid kontrollitud, suundub informatsioon töölauale. Seal saavad kõik komponendid automaatselt ligipääsu neile ettenähtud andmetele ning erinevad vaated, graafikud ja elemendid uuenevad vastavalt.

## 3.2 Andmebaas

Selle töö kaigus valminud andmebaas on seadistatud ettevõttes kasutusolevas Microsoft SQL Serveris. Andmebaas koosneb kahest tabelist (vt. Joonis 8):

- *measures* tabelis on järgmised tulbad:
  - *id* – andmebaasi poolt automaatselt määratud identifikaator;
  - *detected\_at* – ajahetk, millal tootmistakt tuvastati;
  - *ean* – enim tuvastatud triipkood ehk mis toodet parasjagu toodeti;
  - *placement\_type* – mis paigutustüüp oli kasutusel (kas taktis kokku 8 või 12 toodet).
- *results* tabelis on järgmised tulbad:
  - *id* – andmebaasi poolt automaatselt määratud identifikaator;
  - *measure\_id* – viide vastavale tootmistaktile *measures* tabelis;
  - *product\_index* – kus vastav toode kaadris paiknes (kas näiteks vasaku kaamera üleval vasakul või parema kaamera all paremas nurgas);
  - *upper\_label* – kas vastavalt tootelt tuvastati oodatud ülemine silt või mitte;
  - *lower\_label* – kas vastavalt tootelt tuvastati oodatud alumine silt või mitte;
  - *date\_correct* – kas tootelt tuvastatud kuupäev oli oodatud tulemus või mitte;
  - *date* – mis kuupäev vastavalt tootelt tuvastati, kui üldse tuvastati;

- *product\_correct* – kas pakendis oleva toode on olemas/vastab ootustele;
- *product* – mis tootena pakendis olev toode tuvastati;



**Joonis 8.** Loodud andmebaasi relatsiooniline diagramm

Sellise struktuuriga minimaalne andmebaas võimaldab andmeid grupeeritult tagasüsteemi edastada nii, et iga tootmistakti puhul on alati teada iga pakendi kohta kõikide tuvastuste tulemused. Näiteks saab teoreetiliselt teha selliseid päringuid, mis tagastavad tulemusi ainult parema kaamera vasakus ääres olnud toodete kohta. See on kliendile kasulik võimalus, kuna kui töölaualt on näha, et viimaste tootmistaktide jooksul on just sealt toodetelt tuvastatud kuupäevad kahtlaselt ebakorrektsed, siis on võimalik need tootmistaktid töölaual justkui reaalajavaates kuvada. Ühtlasi on selline ülesehitus mälu kasutuse mõttes odav, kuna kõik andmed on kas tõeväärtused, lühikesed sõned või üksikud numbrid.

### 3.3 Töölaud

Selle töö käigus valminud töölaual on kolm vaadet – reaalajas jälgimine, analüüs ning ülevaade. Järgnevalt on kirjeldatud igaüks neist lähemalt. Vaadet saab valida ekraani üleval servas. Kuna töölaud on interaktiivse olemusega ning kuvatõmmistega on liialt keeruline

kõiki funktsionaalsusi kujutada, siis on Google Drive kaustast<sup>15</sup> võimalik leida demonstratiivseid videoid.

### 3.3.1 Reaalajas jälgimine

See vaade võimaldab kasutajal näha reaalajas kõiki tuvastusi, mis kvaliteedikontrolli jooksul tehti. Iga tootmistakti puhul on näha sellel hetkel tootmisliinil olnud toote kohta üldine informatsioon ehk tootenimi, sellega seotud triipkood, toodete paiknevuse tüüp ning aegumiskuupäev ehk kuupäev, mida kvaliteedikontroll pidi õigeaks lugema.

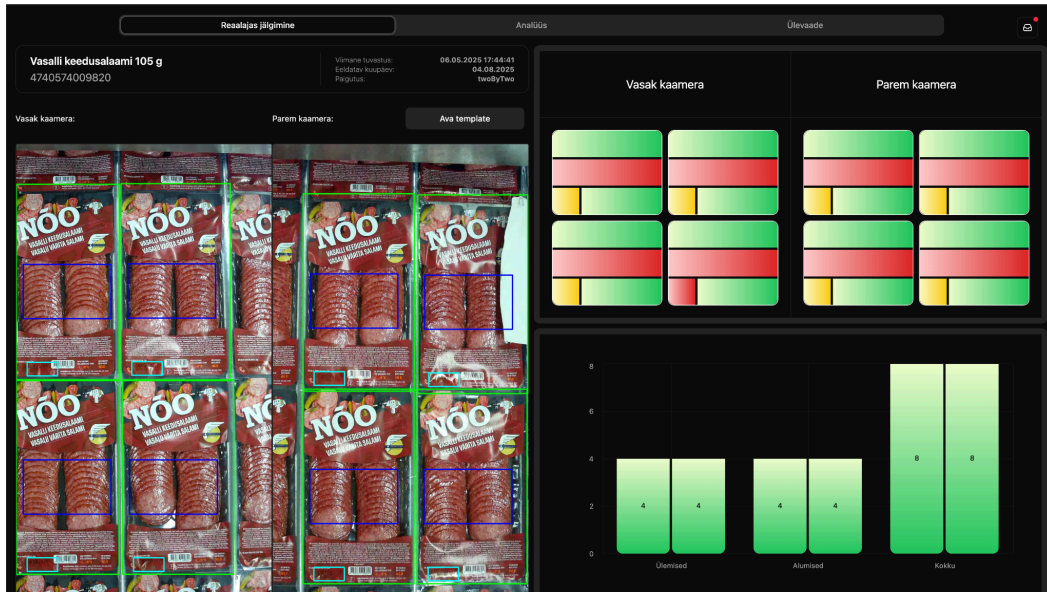
Väiksemateks tükideks jagatud värviliste kujundite abil on võimalik kiiresti aru saada, kui korrektsed nähtud tooted olid. Eraldatud on ülemine ja alumine silt, kuupäev ning tooteaken. Roheline värv kujutab korrektset ning punane ebakorrektselt tuvastust. Kollane värv näitab, et kuupäeva või toodet ei leitud. Lisaks on võimalik kursoriga liikudes näha tuvastusi veel täpsemalt ehk kindla toote puhul, mis tooteks tootetuvastus selle klassifitseeris ning mis kuupäeva kuupäevatu vastus sellelt leidis.

Vaate vasakul poolel on näha mõlema kaamera poolt tehtud jäädvustus, mille põhjal tuvastused tehti ning näha alad, kus neid piltidelt otsiti. Kuna ühe nupuvajutuse kaugusel on ka antud toote mall (*template* ehk eeldatud välimus), siis nende kolme pildi põhjal on võimalik kliendil mõista, kust mingid tuvastusi mõjutavad erisused tekkisid. Näiteks on kohe näha, kui kuupäevi/tooteaknaid otsitakse valedest kohtadest või tooted on üldiselt eeldatava paigutusega võrreldes nihkunud.

Sildituvastuse tulemused on kujutatud tulpdiagrammil, kus on näha ülemiste ja alumiste siltide tuvastuste korrektsused eraldi kaamerate ning terve tootmistakti puhul. Oodatud tuvastused moodustavad rohelised ning mitteodatud punased tulbad.

---

<sup>15</sup> [https://drive.google.com/drive/folders/1jyIP-EDH71wJ3\\_kLkZZSl60x6T3IPk3L?usp=sharing](https://drive.google.com/drive/folders/1jyIP-EDH71wJ3_kLkZZSl60x6T3IPk3L?usp=sharing)



**Joonis 9.** Kõik seni kirjeldatud komponendid töölaua reaalaja vaates. Vasakul on näha hetkel tootmisliinil oleva toote informatsioon ja selle all mõlema kaamera jäädvustused. Paremal on näha üleval kõiki tehtud tuvastusi ning all tulpdiaagrammina sildituvastuse tulemused.

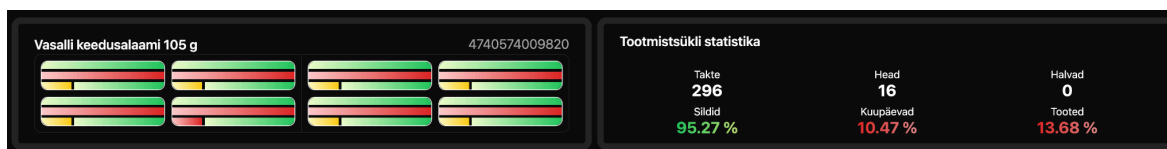
Kuupäeva ja toote tuvastuse tulemusi kujutavad tulpdiaграммид (vt. Joonis 10), kus iga tootmistakti puhul on näha viis enim leitud kuupäeva või toodet. Diagrammide päises on näha ka oodatud tuvastuse osa protsendina, mis on vastavalt kliendi poolt määratud lävenditele kas roheline või punane.



**Joonis 10.** Kuupäeva- ja tootetuvastuse tulemusi kujutavad tulpdiaграммид.

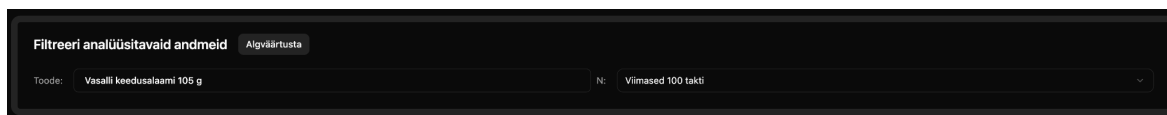
### 3.3.2 Analüüs

See vaade võimaldab analüüsida kindla arvu tootmistaktide kvaliteedikontrolli tulemusi. Vaate üleval osas (vt. Joonis 11) on näha teavet hetkel toodetava toote kohta, ülevaatlikku visualisatsiooni viimati tuvastatud tootmistakti kohta ning statistikat käimasoleva tootmistsükli kohta.



**Joonis 11.** Analüüsi vaate päis. Vasakul on ülevaade reaajas toimuvast, paremal käimasoleva tootmistsükli statistika.

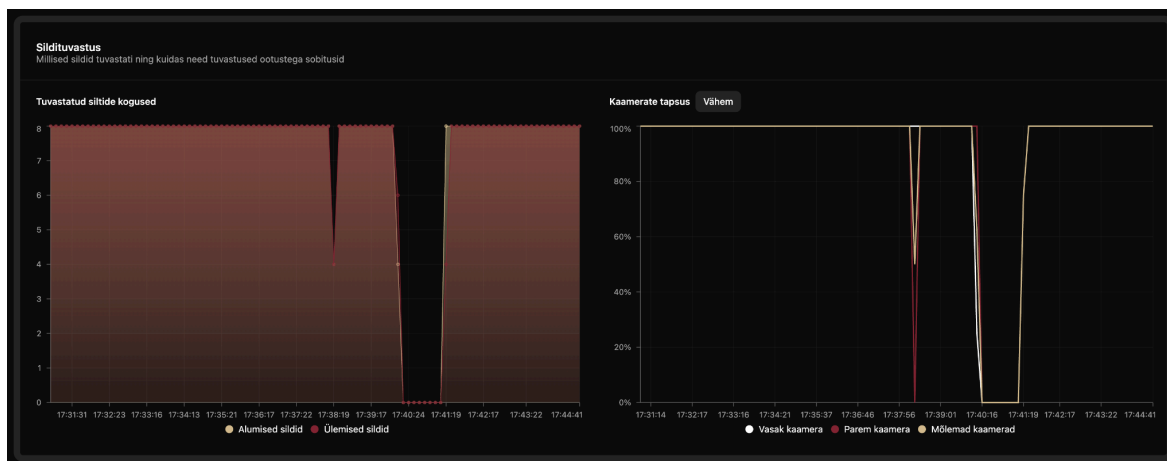
Päise all asuvad filtrid (vt. Joonis 12), millega kasutaja valib toote ning kui mitme tootmistakti kohta ta andmeid soovib näha. Vaikeväärtustena on vaadeldav toode see, mis sellel hetkel tootmisliinil on, kuid valida saab kõigi toodete vahel, mis andmebaasist saadaval on. Vaadeldavate taktide arv on algselt 100, kuid vahetada saab ka 10, 25 või 50 peale.



**Joonis 12.** Päise all olevad filtrid.

Sildituvastuse tulemuste jaoks on kaks joondiagrammi (vt. Joonis 13):

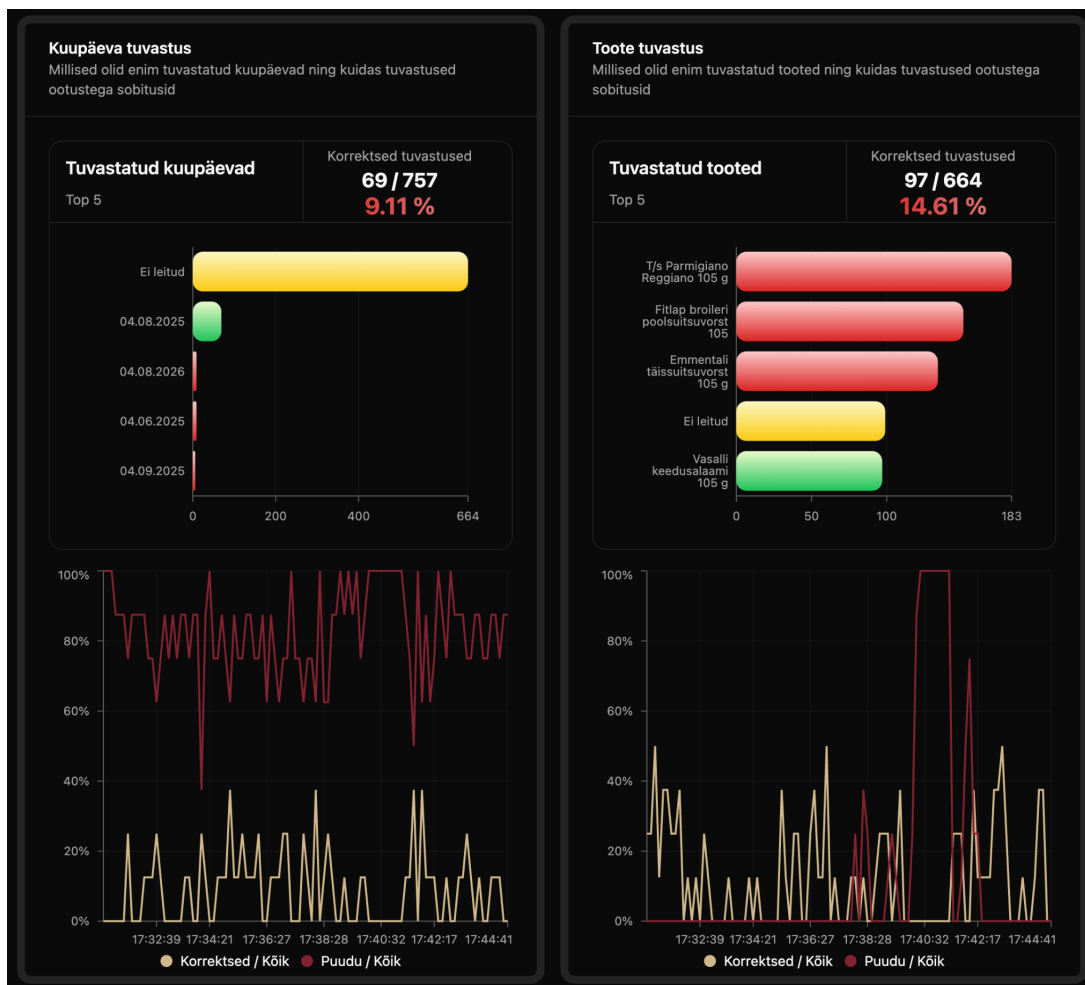
- Esimesel on iga tootmistakti kohta näha kahe joone näol, mitu ülemist ja alumist eeldatud silti tuvastati (12 tootega paigutuse puhul ainult üks joon “ülemiste siltide” tuvastuse kohta).
- Teisel on iga tootmistakti kohta näha kolme joone näol, mitu eeldatud silti tuvastati vasaku kaamera, parema kaamera ja mõlema kaamera jäädvustuste peale kokku.



**Joonis 13.** Sildituvastuse tulemuste graafikud

Nii kuupäeva- kui ka tootetuvastuse tulemuste jaoks on:

- Sama tüüpi tulpdiaagrammid enim tuvastatud kuupäevade/toodete kohta mis reaalaajas, kuid andmestik on ühe tootmistakti asemel kasutaja poolt määratud viimased N tootmistakti (vt. Joonis 14).
- Joondiagramm, kus üks joon näitab oodatud tuvastuste ning teine puuduvate tuvastuste osa kõikide tuvastuste hulgast (vt. Joonis 14).



**Joonis 14.** Kuupäeva- ja tootetuvastuse tulp- ja joondiagrammid

### 3.3.3 Ülevaade

See vaade võimaldab kasutajal otsida andmebaasist kõiki selliseid tootmistakte ning nendega seonduvaid tulemusi, mis vastavad tema poolt määratud tingimustele (vt. Joonis 15). Valida saab järgmiste tingimuste vahel:

- kindel EAN13 kood ehk toode;
- toodete paigutus ehk kas 8 toodet on kahes reas ja neljas tulbas, või 12 toodet on kolmes reas ja neljas tulbas;
- kindel nädalapäev;
- kindel ajavahemik (kuupäevad ja kellaajad);
- minimaalne ja/või maksimaalne sildituvastuse oodatud siltide osa tootmistaktis;
- minimaalne ja/või maksimaalne oodatud kuupäevade tuvastuste osa tootmistaktis;
- minimaalne ja/või maksimaalne oodatud toodete tuvastuste osa tootmistaktis.

Reaalajas jälgimine      Analüüs      Ülevaade

Otsi spetsiifilisi tootmistakte

Toode:  Paigutus:

Alguskuupäev:  Lõppkuupäev:  Nädalapäev:

Sildi tuvastuse min. %:  Sildi tuvastuse max. %:

Kuupäeva tuvastuse min. %:  Kuupäeva tuvastuse max. %:

Toote tuvastuse min. %:  Toote tuvastuse max. %:

Rakenda filtrid      Tühjenda filtrid

**Joonis 15.** Algselt laetud ülevaate leht

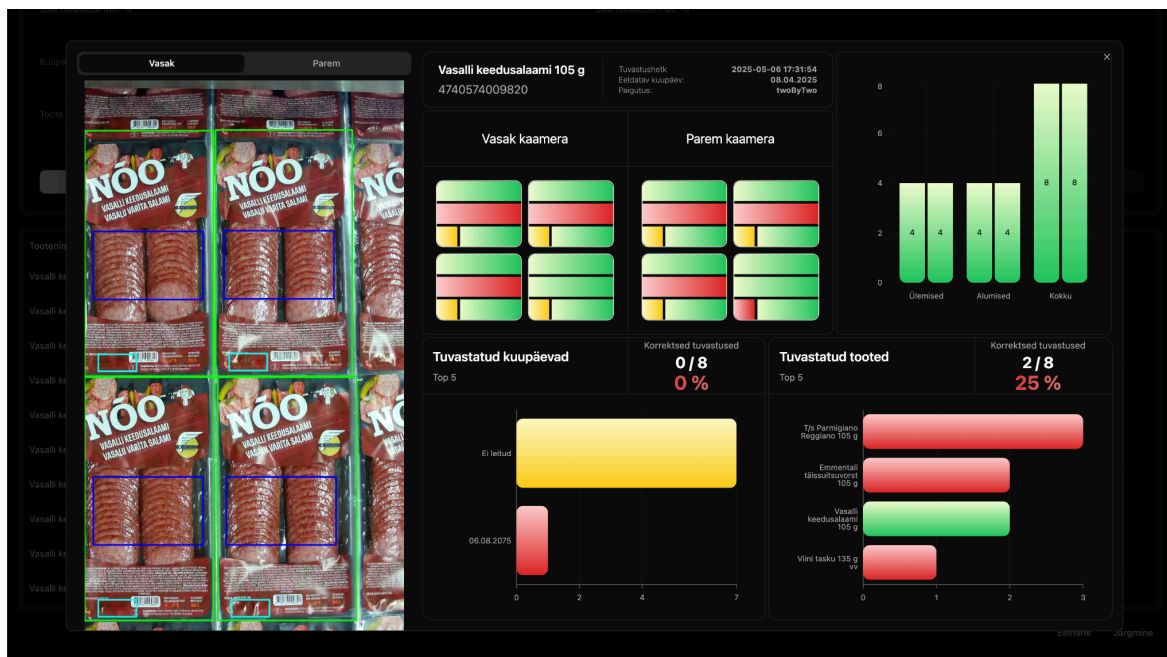
Kui kasutaja on sisestanud talle sobiva filtrite kombinatsiooni, siis kõik valitud tingimustele vastavad tootmistaktid kuvatakse tabeli formaadis ekraanile (vt. Joonis 16).

Tootenimi	EAN	Tuvastatud	Eelvaade
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:51:43	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:49:47	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:49:38	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:49:30	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:49:21	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:49:12	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:48:58	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:48:48	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:48:38	🔍
Vasalli keedusalaami 105 g	4740574009820	2025-05-06 17:48:29	🔍

Eelmine      Järgmine

**Joonis 16.** Kasutaja sisestatud tingimustele vastavad tootmistaktid

Iga andmerea puhul on näha, mis tootele tootmistakt vastab ning millal takt tuvastati. Lisaks on igal real nupp, mis avab hõljukakna (ingl *popover*), kus on näha selle tootmistakti kohta kõik informatsioon, mida oleks olnud võimalik reaalaja vaates näha (vt. Joonis 17).

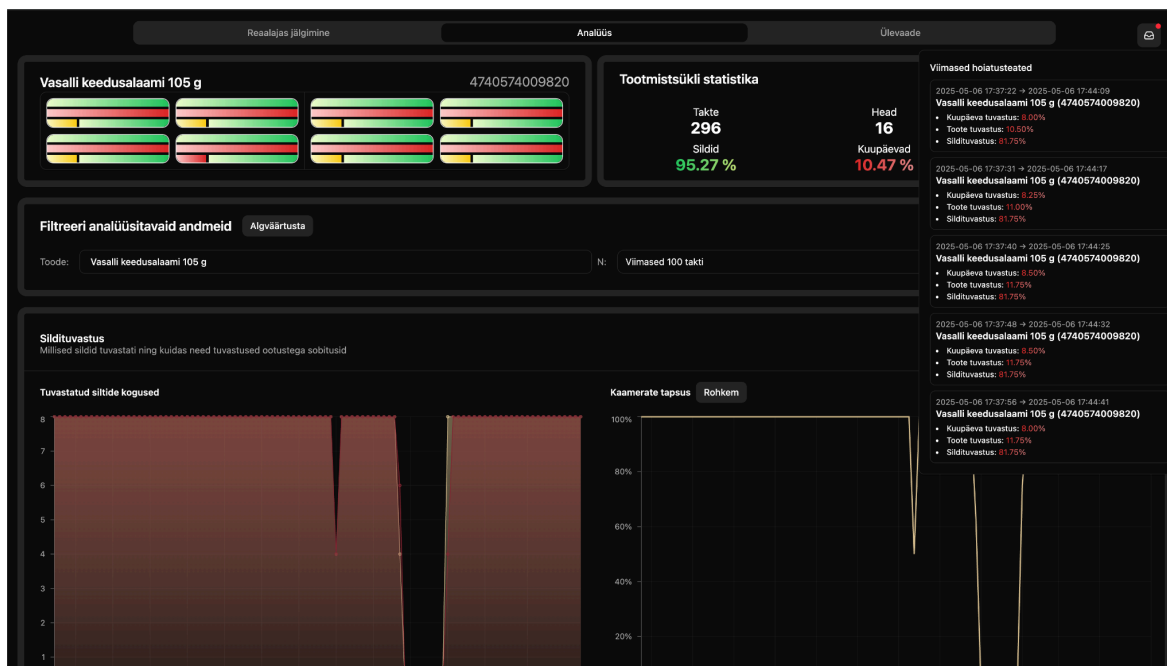


Joonis 17. “Eelvaade” nupule vajutades avanenud aken kindla tootmistakti kohta

### 3.4 Teavitused lävenditest kõrvalekalletest

Olles kursis AIRE projekti käigus valminud kvaliteedikontrolli süsteemi tulemuste usaldusväärsuse ehk täpsusega, määrati kliendi ja töö autori koostöös iga tuvastuskategooria puhul lävend, mille suhtes komponente kas “positiivselt” või “negatiivselt” kujutada. Näiteks analüüsi vaate päises on näha hetkel käimasoleva tootmistsükli kohta üldist statistikat, kus tuvastustega seotud protsendid on kas punased või rohelised. Esimesel juhul on see tuvastus olnud piisavalt pikalt ootustest erinev, teisel juhul mitte. Lävendeid jälgitakse toote põhisel, mitte üle kogu toodangu, ja see toimib Redise vahemälu abil.

Kui tootmine on piisavalt kaua olnud lävendist madalama kvaliteediga, siis loob tagaüstem teavituse juhtunu kohta. Töölaua üleval paremas nurgas on teavituste ikoon, millele vajutades kuvatakse kasutajale viimased viis teavitust (vt. Joonis 18).



**Joonis 18.** Viimased viis töölauale jõudnud lävendist kõrvalekalde teavitust (ülevaate paremal nurgas)

See komponent on kasutatav iga vaate puhul. Kindlale teavitusele vajutamine suunab kasutaja ülevaate lehele ning filtrid on eeltäidetud nii, et vajutades nupule “Rakenda filtrid” kuvatakse kasutajale kõik tootmistaktid, mis selle teavitusega seotud olid.

### 3.5 Kasutuse stsenaariumid

Esmase kasutuse alusel Nõo lihatööstuse tegevjuhi poolt on võimalik kirjeldada loodud lahenduse tegelikku kasutust. Stsenaariumite kohta on demonstratiivsed videod Google Drive kaustas<sup>16</sup>.

**Stsenaarium 1** - Kasutajal on veebibrauseris avatud töölaud. Sellel hetkel tootmisliinil oleva toote puhul on tootetuvastuse täpsus ehk vastavus ootustega piisavalt kaua olnud kvaliteedilävendist madalam, seega tagasüsteem saadab töölauale teavituse. Kasutaja märkab seda ning avab selle, nähes et toote X puhul on tootetuvastuse täpsus Y% ning teavituse tekitanud ajavahemik on AJAHETK1-AJAHETK2. Teavitusele vajutades suunatakse kasutaja ülevaate aknasse, kus filtrid ja tunnused on tema eest juba täidetud. Vajutades nupule “Rakenda filtrid”, kuvatakse kasutajale kõik need tootmistaktid, mis lävendist kõrvalekaldega

<sup>16</sup> [https://drive.google.com/drive/folders/1jyIP-EDH71wJ3\\_kLkZZSl60x6T3IPk3L?usp=sharing](https://drive.google.com/drive/folders/1jyIP-EDH71wJ3_kLkZZSl60x6T3IPk3L?usp=sharing)

seotud olid. Iga tulemuse puhul on tal võimalik näha selle takti kohta seda informatsiooni, mis oleks reaalaraja vaates nähtav olnud.

**Stsenaarium 2** - Kasutajal on veebibrauseris avatud töölaud. Analüüsi vaate päises on näha, et äsja töölauale jõudnud takti puhul on iga tuvastuskategooria puhul tulemused ootustest täiesti erinevad. Kasutaja suundub reaalarajavaatesse ning uurib tuvastusi. Uus takt kirjutab informatsiooni üle, kuid kasutaja teab täpset ajahetke, millal tagasüsteem probleemse tootmistaktiga tegutses. Kasutaja suundub ülevaate lehele ning otsib nähtud toote takti, mille kvaliteeti kontrolliti täpselt sellel ajahetkel. Vajutades nupule “Rakenda filtrid”, kuvatakse kasutajale äsja nähtud tootmistakt ning tal on võimalik selle takti kohta näha seda informatsiooni, mida ta hetk tagasi nägi reaalaraja vaates.

### **3.6 Kliendi tagasiside**

Klient oli tehtud tööga väga rahul. Kuna kõik nõuded ja ideed olid algusest peale paika pandud töö autori ja kliendi koostöös, siis otseseid puudujääke ei tuvastatud. Kasutajaliides oli piisavalt loogiline ja informatiivne, et kõik funktsionaalsused said selgeks õpitud sekunditega. Samuti jälgiti töötavat lahendust mõnda aega ning kvaliteedikontrolli suhtes sai klient kiiresti aru, kust tekkisid erisused ning tootmisprotsessil silma peal hoida nii reaalarajas kui ka ajaloolise vaate ja filtreeritud andmestiku läbi.

### **3.7 Potentsiaalsed edasiarendused**

Kuigi selle töö käigus loodud süsteem pakub kasutajale mitmeid funktsionaalsusi, on võimalik seda veelgi edasi arendada. Näiteks kui kasutaja otsib ülevaate lehelt kindlate tunnustega tootmistakte, siis saaks kasutajale kuvada analüüsi vaate, kus kõik graafikud ja diagrammid kujutavad just nende andmete kokkuvõtet. Praegu saab ülevaate lehelt vaadata iga takti tulemusi ühe kaupa.

Palju tähtsat informatsiooni toodete ja üldise konfiguratsiooni kohta on endiselt määratud ja salvestatud AIRE projekti raames valminud süsteemis. Näiteks on seal toodete paigutustega seotud koordinaadid üheselt määratud, kuid tooted liinil on vahel nende eeldatud asukohtade suhtes nihkes ning neid asukohti saab muuta ainult siis, kui kvaliteedikontrolli süsteem on peatatud. Oleks kasulik, kui ROI-de (ingl *region of interest*) defineerimine asuks hoopis selle töö käigus valminud eesliideses, kuna siis saaks neid parameetreid jooksvalt muuta.

Valminud lahendus ei võimalda uurida konkreetseid tootmistsükkeid. Kliendile oleks ka see väga kasulik funktsionaalsus, kuna võib tekkida olukord, kus toote X tsükli lõpus on näha, et tuvastustulemused on korduvalt lävenditest kõrvale kaldunud ning see võimaldaks järgmise tsükli korral nende lõpptulemusi võrrelda. Selle abil saaks ka iga tootmistsükli lõppedes, päeva õpus või mis iganes perioodi kohta koostada ülevaatliku raporti. Selliste automaatsete raportide loomine polnud käesoleva töölaua arenduse loomulik osa, kuid liinil jooksvaid triipkoode jälgides peaks olema lihtne partiide piirid tuvastada ning seejärel andmed filtreerida (sarnaselt ülevaate režiimis tehtud filtreerimisele).

Hetkel on kõik kaamerate jäädvustused salvestatud Cloudinary pilvehoidlasse, kuid lõpplahenduse testimise käigus sai klient aru, et eelistab pikemas perspektiivis lokaalset lahendust. Seda just seetõttu, et suhtlus käib hetkel API-de kaudu ehk võrguliikluse tihedus ja maht on suurem kui sisevõrgus paikneva andmebaasi puhul.

Kliendi tulevikusuund näeb ette kvaliteedikontrolli süsteemi laienemist ka teistele tootmisliinidele ehk töölaud vajab selle funktsionaalsuse toetamiseks teistsugust loogikat. Seda näiteks MPA ja SPA kombineerimisel – URL näitab, mis tootmisliini andmeid töölaud kasutab ning sisu on endiselt SPA. Muudatusi tuleks teha ka terves tagasüsteemis, sest andmed hakkaksid tulema erinevatest allikatest ning neid tuleks eraldi hoiustada ning töödelda.

## Kokkuvõte

Antud bakalaureusetöö eesmärgiks oli luua veebipõhine töölaud ja andmebaas olemasolevale kvaliteedikontrolli süsteemile, mis arendati AIRE demoprojekti raames koostöös Nõo lihatööstusega. Varasem süsteem võimaldas tulemuste kuvamist ainult reaalajas, fikseeritud ja kohandamisvõimaluseta jooniste abil ning andmeid ei salvestatud.

Esmalt selgitati kliendi ja töö autori koostöös välja kasutajate vajadused ja pandi paika funktsionaalsed ja mittefunktsionaalsed arendusnõuded. Ühtlasi arutati, mis võiks alles jääda varasemast lahendusest ning mida tuleks teha teisiti. Töö käigus uuriti töölaudade ja andmebaaside tehnoloogiaid ning tavasid.

See töö lisas süsteemile püsiva andmesalvestuse, rohkelt analüüsivõimalusi ning kasutajasõbralikuma töölaua. Lõpptulemuseks oli Microsoft SQL Serveri abil toimiv relatsiooniline andmebaas ning Reactil põhinev kolme vaatega veebipõhine töölaud, mis võimaldab kliendil mõista tootmises aset leidvaid erisusi, jälgida ja analüüsida toodangut erinevate ajaperioodide ja toodete lõikes ning reageerida lävenditest kõrvalekalletele kiirelt ja tõhusalt.

## Viidatud kirjandus

- [1] Yigitbasioglu OM, Velcu O. A review of dashboards in performance management: Implications for design and research. *International journal of accounting information systems*. 2012 Mar 1;13(1):41-59.
- [2] Vershinin IS, Mustafina AR. Performance analysis of PostgreSQL, MySQL, microsoft SQL server systems based on TPC-H tests. In 2021 International Russian Automation Conference (RusAutoCon) 2021 Sep 5 (pp. 683-687). IEEE.
- [3] Sarikaya A, Correll M, Bartram L, Tory M, Fisher D. What do we talk about when we talk about dashboards?. *IEEE transactions on visualization and computer graphics*. 2018 Aug 21;25(1):682-92. (14.05.2025)
- [4] Orts D. *Dashboard development and deployment*. Bellevue: Noetix Corporation. Bellevue: Noetix Corporation. 2004. (14.05.2025)
- [5] Haan K. The best data Visualization Tools. *Forbes Advisor*. 2024. Available from: <https://www.forbes.com/advisor/business/software/best-data-visualization-tools/> (14.05.2025)
- [6] Krishnan V. *Research data analysis with power bi*.
- [7] Huang H, Ravi S, Warrington T, Cui H, Wang C, McCreary M, Lauffer B, Lu C. A comparison of data visualization tools: A case study in health-related research. *Information Visualization*. 2025 Jan;24(1):62-78.
- [8] GeeksforGeeks. Power BI vs Tableau – Which one is Right for You? *GeeksforGeeks*. 2024 Sep 25; Available from: <https://www.geeksforgeeks.org/power-bi-vs-tableau/>
- [9] MUS M. Comparison between SQL and NoSQL databases and their relationship with big data analytics. *Asian Journal of Research in Computer Science*. 2019.
- [10] Faroult S, Robson P. *The art of SQL*. " O'Reilly Media, Inc."; 2006 Mar 10.
- [11] Vaish G. *Getting started with NoSQL*. Packt Pub.; 2013 Mar 26.
- [12] Mitra A. *Fundamentals of quality control and improvement*. John Wiley & Sons; 2016 May 2.
- [13] ai-robotics-estonia/2024\_AI\_Powered\_Quality\_Control\_Noo\_Lihatoostus. *GitHub*. Available from: [https://github.com/ai-robotics-estonia/2024\\_AI\\_Powered\\_Quality\\_Control\\_Noo\\_Lihatoostus?tab=readme-ov-file](https://github.com/ai-robotics-estonia/2024_AI_Powered_Quality_Control_Noo_Lihatoostus?tab=readme-ov-file) (14.05.2025)
- [14] Elke. Nõo Lihatoostus rakendas Eestis esimesena tehisintellekti pakendite kontrollimiseks - Nõo. Nõo Lihatoostus. 2024. Available from: <https://noo.ee/uudised/noo-lihatoostus-rakendas-eestis-esimesena-tehisintellekti-pakendite-kontrollimiseks/> (14.05.2025)
- [15] rwestMSFT. What is SQL Server? - SQL Server. *Microsoft Learn*. Available from: <https://learn.microsoft.com/en-us/sql/sql-server/what-is-sql-server?view=sql-server-ver16> (14.05.2025)
- [16] Rad BB, Bhatti HJ, Ahmadi M. An introduction to docker and analysis of its performance. *International Journal of Computer Science and Network Security (IJCSNS)*. 2017 Mar 1;17(3):228.
- [17] Ghimire D. Comparative study on Python web frameworks: Flask and Django.
- [18] Carlson J. *Redis in action*. Simon and Schuster; 2013 Jun 17.
- [19] Cloudinary. Image and video upload, storage, optimization and CDN. *Cloudinary*. 2025. Available from: <https://cloudinary.com/> (14.05.2025)
- [20] Phan K. *A Comprehensive Study on Single-Page Applications: Pros, Cons, and Practical Guidelines*.
- [21] Development D. The evolution of React. *HackerNoon*. 2018. Available from: <https://hackernoon.com/the-evolution-of-react-48409fac2efd> (14.05.2025)
- [22] Technology | 2024 Stack Overflow Developer Survey. Available from: <https://survey.stackoverflow.co/2024/technology#most-popular-technologies-webframe-prof> (14.05.2025)

# Litsents

## **Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks**

Mina, **Harmo-Ron Kallaste**,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose  
  
Tootmisliini kvaliteedijälgimise süsteemi andmebaasi ning töölaua loomine, mille juhendaja on Ardi Tampuu, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.
2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

**Harmo-Ron Kallaste**

14.05.2025