

Tehisarupõhine tarkvaraarendus ja agentide juhtimine algajatele

Töövihik

Linux

Pi ja TalTechi hpc-vllm Linuxis



Eesmärk / Piirid / Väike samm
Kontroll / Otsus / Jätkamine

Sisukord

Töövihiku kasutamine	2
1. Seadistus: paigalda Pi, loo võti ja vali hpc-vllm	3
Node.js ja Pi eeldused	3
Pi paigaldus ja isiklik võti	4
hpc-vllm seadistus	5
Võtme sisestamine ja ligipääsu kontroll	6
2. Loomine: anna väike ja selge ülesanne	7
Täpsustamine ja loomiskäsk	8
3. Kontroll: vaata tulemust brauseris	9
4. Parandus: kirjelda täpselt üht viga	10
5. Taastamine: lõpeta töö enne koopia tagasipanekut	11
6. Uus seanss: anna üle kontrollitud tööseis	12
Kui tulemus on teistsugune	13
Enesekontroll ja kontrollitulemuste piirid	13
Kirjandus ja juhendid	13
Paberil kasutatavad mallid	14
Pi hpc-vllm täielik models.json näide	14
Loomise juhised	15
Paranduse juhised	15
Projektimärkme mall	16
Uue seansi juhised	16

Töövihiku kasutamine

See on õpiku „**Tehisarupõhine tarkvaraarendus ja agentide juhtimine algajatele**” eraldi praktiline töövihik TalTechi õppurile Linuxis. Õpik selgitab põhimõtteid; siin rakendad neid Pi ja kooli hpc-vllm teenusega.

Selles töövihikus juhid agenti, kes teeb brauseris töötava matkavarustuse loendi. Sina kirjeldad soovi ja kontrollid tulemust; koodi lugemist ega käsitsi parandamist ei nõuta. Töö käib kuue sammuga: **seadistus** → **loomine** → **kontroll** → **parandus** → **taastamine** → **uus seanss**.

Vajad Linuxi arvutit, Bash-terminali, brauserit, Pi-d (pi.dev) ja lubatud ligipääsu TalTechi AI-proxy kaudu hpc-vllm teenusele. Kasutad enda õppurikonto võtit, mitte kellegi teise võtit. Kooliligipääs ei ole kõigile iseõppijatele automaatselt olemas. Kontrolli enne konto loomist ja teenuse kasutamist kooli, Pi ning oma konto kehtivaid kasutus- ja vanusetingimusi; hpc-vllm valik üksi ei tõenda, et kõik vanused võivad teenust kasutada.

Paberilt lugedes on kõik seadistuse põhiaandmed, kontrollid ja näited siinsamas. PDF-i lõpus „Paberil kasutatavad mallid” on täielik JSON ja neli täidetavat tekstimalli; digitaalses juhendis saad need eraldi failina. Digitaalsed failid on kopeerimise mugavus; paberilt saad arutada ka veanäidet ilma koodi lugemata. Paber üksi ei käivita Pi-d ega loo brauserifaili; päris agendikatse vajab arvutit, ligipääsu, piiratud keskkonda ja eraldi luba.

PDF-i lugemine ja mallide täitmine ei vaja internetti. Pi paigaldus, konto seadistus ja hpc-vllm mudelipäringud vajavad võrku. Rakenduse kohalik kasutus ja selle arendamine kaugmudeliga on eri asjad: teenusele võivad jõuda juhis, loetud failisisu ja tööriistatulemused. Mallides on keelatud rakenduse võrguühendused; arendamiseks kasutatakse ainult kokkulepitud mudeliteenust.

Abifailid. Soovitan luua õppekausta ise failid alamkausta ning salvestada sinna töövihikuga saadud näidisfailid ja enda täidetud mallid. Kui näidisfaile pole kaasas, küsi neid töövihiku andjalt: tühi kaust neid ei loo. See pole agendi töökaust ega kaitstud taastekoopia.

Veebilingid: Pi <https://pi.dev>; TalTechi AI-proxy <https://ai-proxy.cm.itcollege.ee/>.

Samu projektifaile saab kasutada eri AI-vahenditega, kuid ühine kaust ei ole ühine vestlusmälu. Algaja kasutab üht muutvat vahendit korraga.

Enne agenditööd: vaja on kontrollitud failiõiguste, eelarve ja täieliku peatamise korraldust. Tühi töökaust ega juhis ei ole turvaliivakast. Kui neid piire sinu keskkonnas pole, piirdu valmisfailide kontrollimisega; see harjutab tulemuse vaatlemist, mitte agendiga loomist.

1. Seadistus: paigalda Pi, loo võti ja vali hpc-vllm

Seos õpikuga: peatükid 1–2 ja 10. Pi on agenti käitav töövahend, mitte mudel ega mudeliteenus. TalTechi AI-proxy vahendab ligipääsu hpc-vllm teenusele; Qwen3.8-27B on selles juhendis kasutatav mudel.

Node.js ja Pi eeldused

Paigalda Pi enne võtme seadistamist. Terminal on aken, kuhu sisestad käske. Kui Pi on juba olemas, kontrolli ainult selle versiooni ning jäta paigaldamine vahele. Muul juhul veendu, et kasutad arvutit, kuhu sul on lubatud tarkvara paigaldada, ja et tead allalaadimise allikat. Pi juhendis on Pi jaoks kaks paigaldusviisi: npm (vajadusel paigalda enne Node.js nvm-iga) või Pi ametlik paigaldaja. Vali **üks Pi paigaldusviis**.

1. Kontrolli Node.js-i ja npm-i terminalis:

```
node --version
npm --version
```

npm-i tee vajab Node.js-i versiooni **22.19 või uuemat** ja npm-i. Kui mõlemad on olemas ning versioon sobib, jätk punktist 3. Kui need puuduvad või Node on liiga vana, saad oma Linuxi kasutajakontole Node.js-i paigaldada nvm-iga (*Node Version Manager*), kui arvuti reeglid seda lubavad. See ei vaja süsteemiüleseid administraatoriõigusi. Kui nvm on juba olemas, jäta selle paigaldamine vahele ja jätk punktist 2.

2. Valikuline eelsamm npm-i teele: paigalda Node.js nvm-iga. Autori eestikeelne Pi paigaldusjuhend kasutab nvm-i versiooni v0.40.3. Paberilt sisestatav lähteadress on <https://github.com/nvm-sh/nvm> (versioon v0.40.3); allolev käsk kasutab selle versiooni `install.sh` faili. Enne käivitamist veendu, et allikas, versioon ja arvuti kasutusreeglid sobivad sulle; vajadusel küsi haldajalt abi. **Käsk laadib võrgust skripti ja käivitab selle kohe** ning võib muuta sinu shelli käivitusfaili. Ära käivita seda pimesi ega lisa `sudo`.

```
curl -fsSL https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.3/install.sh | bash
```

Ava uus Bash-terminal või laadi oma kasutaja Bash-seadistus uuesti; Zsh-i puhul ava uus Zsh-terminal või laadi selle seadistusfail:

```
source ~/.bashrc
nvm --version
```

Kui kasutad Zsh-i, asenda `source ~/.bashrc` käsuga `source ~/.zshrc`. Kui nvm ei käivitu, peatu ja kontrolli koos haldajaga, kas paigaldus ning sinu shelli käivitusfail õnnestusid; ära muuda süsteemifaili pimesi. Seejärel paigalda Node.js-i pika toega (LTS) väljalase ning kontrolli tegelikku versiooni (LTS-i nimetus üksi ei taga Pi miinimumi):

```
nvm install --lts
node --version
npm --version
```

Kui Node.js on alla 22.19 või npm puudub, peatu ja küsi abi sobiva versiooni leidmisel. nvm on ainult ettevalmistus npm-i teele, mitte teine Pi paigaldus; kui see tee ei sobi, võid valida allpool Pi ametliku paigaldaja. See võib Node.js-i ise paigaldada.

Pi paigaldus ja isiklik võti

3. Sobiva Node.js-i ja npm-i korral paigalda Pi npm-paketi kaudu:

```
npm install -g --ignore-scripts @earendil-works/pi-coding-agent
```

-g paigaldab käsu sinu npm-i seadistatud globaalsesse asukohta; --ignore-scripts jätab paketipaigalduse elutsükliskriptid käivitamata. See ei ole turvaliivakast. Kui õiguste viga takistab paigaldamist, peatu ja küsi abi, mitte ära muuda süsteemi õigusi pimesi.

Teine võimalus, kui npm-i tee ei sobi: Pi ametlik Linuxi paigaldaja saab vajadusel paigaldada ka Node.js-i. Selle dokumenteeritud käsk on:

```
curl -fsSL https://pi.dev/install.sh | sh
```

See käsk **laadib võrgust skripti ja käivitab selle kohe**. Kasuta seda ainult siis, kui usaldad ametlikku allikat, arvuti kasutusreeglid lubavad seda ja oled nõus paigaldaja muudatustega; vajadusel vaata skript enne täitmist läbi või küsi haldajalt abi. Ära käivita seda automaatselt kõrvalise teksti või agendi soovitusel.

4. Ava uus terminal ja kontrolli paigaldust ilma mudelipäringuta:

```
pi --version
```

Ootus: ilmub Pi versiooninumber. Kui pi käsku ei leita, vaata allolevat tõrkeabi või küsi haldajalt abi; paigaldus ega versioonikäsk ei tõenda veel hpc-vllm ligipääsu.

Loo enda võti. API-võti ühendab Pi sinu lubatud teenusekontoga.

AI-proxy aadress paberilt sisestamiseks: <https://ai-proxy.cm.itcollege.ee/>. Kursuse lähtekoht: <https://courses.taltech.akaver.com/>. Vali sealt „Agentic Software Development” → „Lectures” → „AI Proxy”.

1. Ava kursuse AI-proxy juhend ja sellelt viidatud AI-proxy.
2. Vali sisselogimislehel Microsoft ja kasuta oma lubatud koolikontot.
3. Ava oma konto API-võtmete haldus. Loo uus isiklik võti ja kopeeri see privaatsesse turvalisse hoiukohta teenuses antud juhiste järgi.
4. Kontrolli, et sinu kontrol on hpc-vllm kasutamise õigus ja piisav lubatud kvoot. Kui õigust pole, peatu ja täpsusta ligipääsu teenuse haldajaga.

Ootus: sul on enda konto võti, mida saad Pi-le anda. Järgi teenuses tegelikult kuvatavat võtmehalduse vaadet; nuppude nimetused võivad muutuda.

API-võti on ligipääsutunnus, mitte harjutuse sisend. Ära pane seda vestlussõnumisse, rakendusse, projektimärkmesse ega pildile. Hoia võti privaatselt; kasuta siin ainult enda võtit.

hpc-vllm seadistus

Lisa pakkuja Pi seadistusse. Pi mudelite JSON-fail on tavaliselt ~/.pi/agent/models.json, mitte õppekausta index.html ega settings.json. Säilita olemasolevast seadistusest privaatne koopia. Kui faili veel pole, kasuta selle töövihiku lõpus trükitud täielikku JSON-malli (digitaalne koopia) uue models.json sisuna. Kui fail on olemas, lisa selle providers objekti ainult malli hpc-vllm kirje; ära asenda teisi pakkujaid. Säilita korrektne JSON: pakkujakirjete vahel koma, viimase järel mitte.

Siin on lühike ülevaade; töövihiku lõpus on täielik mall koos Pi ühilduvusseadete ja mõtlemisrežiimi kirjeldusega:

```
{
  "providers": {
    "hpc-vllm": {
      "baseUrl": "https://ai-proxy.cm.itcollege.ee/hpcvllm/v1",
      "api": "openai-completions",
      "apiKey": "$TALTECH_API_KEY",
      "models": [{ "id": "Qwen3.8-27B" }]
    }
  }
}
```

hpc-vllm ühenduse avalik JSON-mall

Lisa kirje Pi models.json-i, säilitades teised pakkujad. Pildil on malli põhiandmed, mitte autori privaatne seadistus.

```
{
  "providers": {
    "hpc-vllm": {
      "baseUrl": "https://ai-proxy.cm.itcollege.ee/hpcvllm/v1",
      "api": "openai-completions",
      "apiKey": "$TALTECH_API_KEY",
      "models": [
        {
          "id": "Qwen3.8-27B"
        }
      ]
    }
  }
}
```

\$TALTECH_API_KEY on viide. Enda võtme sisestad terminalis peidetult; võtit ei ole pildil ega mallis. Täielik fail sisaldab ka Pi ühilduvusseadeid.

Avaliku JSON-malli põhiandmed: võtme asemel on keskkonnamuutuja viide

Pildil on võtme asemel keskkonnamuutuja viide. Teisi pakkujaid sisaldava faili puhul lisad sama kirje olemasolevasse providers objekti. Sa ei pea JSON-i programmeerimiskeelena õppima, kuid pead säilitama jutumärgid, sulud ja pakkujakirjete vahelised komad.

Võtme sisestamine ja ligipääsu kontroll

Anna võti samale terminalile. \$TALTECH_API_KEY on viide keskkonnamuutujale, mitte sinu võti. Järgmine Bash-käsk küsib võtit seda ekraanile ega käsuaajaloosse kirjutamata. Kleebi enda võti küsimise järel:

```
read -r -s -p "TalTechi API-võti: " TALTECH_API_KEY
printf '\n'
export TALTECH_API_KEY
```

Ära asenda käsku võtmega ega kasuta echo, printenv või credentiale väljastavaid käske. Muutuja peab olema olema samas terminalis, kust Pi käivitad; uues terminalis sisesta võti uuesti. See ei ole krüpteeritud hoidla. Hoia seadistusfail ja selle koopia privaatselt, väljaspool jagatavat projekti.

Kontrolli pakkuja võtit kuvamata:

```
pi auth check --provider hpc-vllm --json --no-refresh
pi --list-models hpc-vllm
```

Ootus: ligipääsu vastuses on "status": "ready" ja mudeliloendis pakkuja hpc-vllm, mudel

Qwen3.8-27B. Mudeli nimi on tõstutundlik. ready tähendab leitud ligipääsu, mitte edukat mudelipäringut ega piisavat kvooti. Ära lisa --credentials lippu. Mudelivalikus ei kasutata teisi, eriti tasulisi pakkujaid varuteena.

Kursuselehe järgi on mudel tasuta. Kontrolli oma konto õigust ja kvooti. Makse- või kvooditeate korral peatu; harjutus ei eelda krediidi ostmist.

2. Loomine: anna väike ja selge ülesanne

Seos õpikuga: peatükid 3–5. Päris matkaks võib paberloendist piisata; siin lood rakenduse, et õppida agendi juhtimist.

Loo failihalduris uus tühi õppekaust. Hoia päris projektid, isikuandmed ja saladused sealt väljas. Ava selles kaustas terminal; kui menüüs sellist valikut pole, liigu terminalis `cd` käsuga oma kausta. `pwd` näitab kausta asukohta.

Luba ainult selle töökausta `index.html` loomine; ei paigaldusi, serverit, avaldamist ega teisi faile. Kasuta ainult väljamõeldud esemeid, näiteks `telk` ja `lamp`, ning kooli lubatud kvooti mudeliga `hpc-vllm/Qwen3.8-27B`.

Täida enne **iga** loomise, paranduse või uue seansi käivitust:

Ülesanne ja kuupäev: _____

Mudel / kasutusõigus, kvoot ja kontrolli aeg: _____

Töökaust / lubatud failid, tööriistad ja andmed: _____

Piirid: _____ käivitust, _____ päringut, _____ minutit.

Väljundipiir: _____ tokenit; kulu: 0; tasulist varuteed pole.

Piiride jälgimise ja rakendamise viis: _____

- ☐ Oskan aktiivse töö peatada ja selle lõppu kontrollida (samm 5).
- ☐ Kaitstud taastekoopiast taastamine on proovitud (samm 5).

Piiride rakendamine. Juhises olev arv ei peata tööd automaatselt. Üks käivitus võib teha mitu mudelipäringut, ka automaatsete korduste tõttu. Token on mudeli töödeldav tekstiühik, mitte tingimata sõna.

- **Käivitused:** märgi iga alustatud katse, ka ebaõnnestunud katse.
- **Aeg:** pane enne käivitust taimer käima ja jäta aega kontrollideks. Aja lõppedes katkesta töö sammu 5 järgi ning kontrolli, et see on lõppenud.
- **Päringud ja väljund:** vaja on loendavat ja piiril peatavat valvurit, mis arvestab ka automaatseid kordusi ja väljundtokenite mahtu. Küsi ettevalmistatud lahendust õppekeskkonna haldajalt; taimer ei asenda seda.

Kui sa ei saa piire jälgida ja rakendada, ära alusta mudelikatset; jätka paberharjutustega. Peatu esimese piiri täitumisel, tõrke, kvoodi-/makseteate või lubamatu tegevuse korral. Uus katse vajab uut luba ja eelarvet.

Täpsustamine ja loomiskäsk

Järgmine käsk on Pi avamise **näidis juba piiratud keskkonnas**:

```
pi --provider hpc-vllm --model Qwen3.8-27B --thinking off \  
  --tools write --no-approve --no-extensions --no-mcp \  
  --no-skills --no-prompt-templates --no-context-files \  
  --no-session --offline
```

Loomiseks piisab `write` tööriistast: kaust on tühi ja muudetav fail on üks. Nii ei teki vajadust shellikontrollide ega kõrvaliste failide lugemise järele. Need lipud vähendavad automaatselt laaditavaid lisasid. Need **ei sea** faili-, päringu-, aja- ega tokenipiire. Ka `--offline` ei keela sinu antud mudelipäringut. Ka `--no-session` ei kustuta juba saadetud andmeid. Tehniline piir on lihtne: turvapiir ja eelarvevalvur peavad toimima väljaspool vestlust; kui neid pole, ära alusta agenditööd.

Täpsusta soovi enne teostamist. Näiteks:

Sina: „Tahan väikest matkavarustuse loendit. Ära veel faili muuda. Kas soov on piisavalt selge?”

Agent: „Kas esemete nimed peavad pärast lehe sulgemist alles jääma?”

Sina: „Ei. See on proov sünteetiliste andmetega, ilma salvestuseta.”

See on näidisdialoog, mitte kohustuslik sõnastus ega tegelik katsevestlus. Palu enne teostust lühike nähtav plaan ning lepi kokku kontrollid ja teostusluba; eesmärgi täpsustamine üksi ei anna veel muutmisluba. Seejärel kasuta lõpus trükitud lühikest **loomise juhist** (või selle digitaalset koopiat), täida oma piirid ja sisesta juhis Pi-sse. Agent valib teostuse; sina ei kinnita koodiridu.

Ootus: kausta tekib `index.html`. Agendi „valmis” ei tõenda veel, et rakendus töötab. Kui faili ei teki või töö katkeb, peatu ja säilita tegelik seis.

3. Kontroll: vaata tulemust brauseris

Seos õpikuga: peatükid 3 ja 7. Veendu enne kontrolli, et aktiivne muutmistöö on lõppenud. Märki iga proovi juurde kontrollitud failikoopia, aeg ja tegelik tulemus, mitte üksnes agendi kokkuvõte.

Ava `index.html` failihaldurist või brauseri „Ava fail” käsuga. Serverit ei ole vaja. Kontrolli aadressiribalt, et avatud on õige kohalik fail.

Kui loomine ei õnnestunud, saad paberil võrrelda allolevat tabelit ja selle lehe toimiva näite pilti. Arvutis võid eraldi avada digitaalse toimiva näidisfaili. See on ettevalmistatud näide, mitte sinu agendi tulemus. Pildilt mõtlemine ega faili kopeerimine ei tõenda AI-ga loomist.

Proov	Oodatav tulemus
Ava fail uuesti	Loend on tühi; proovisisu ei säili
Lisa nupuga <code>telk</code>	Tekib üks rida
Lisa Enteriga <code>lamp</code>	Tekib teine rida
Märki ainult <code>lamp</code>	<code>telk</code> jääb märkimata
Lisa tühi või ainult tühikutega sisend	Uut rida ei teki
Lisa teine <code>lamp</code> , märki uus rida	Sama nimega read on eraldi märgitavad
Lisa <code><lamp></code>	Nähtav on see tekst, mitte uus leheelement

Matkavarustuse loend

Märkus: lehel puudub salvestus — lõpetades brauseri või värskendades lehte kaob sisestatud loend.

telk ☐

lamp ☒

Näidisloend: ainult lamp on märgitud

Pilt näitab kahe rea seisu pärast ainult lambi märkimist; see ei ole sinu faili kontrollitulemus. Võrdluseks on näidisfail. Tee kõik tabeli proovid ka oma failiga ise. Automaatkontroll toetab vaatlust, kuid ei ole turva-ega õigusaudit. Proov ei salvesta loendit.

Kohalik asukoht ja õigesti töötav nupp ei tõenda võrguühenduste puudumist, täielikku turvalisust ega materjali kasutusõigust. Vaata üle ka kõrvalised failimuudatused ja puuduva tõendiga nõuded; märki teadmata kohad ausalt.

Otsus: kui kõik proovid läbivad, säilita failihalduriga toimiv koopia agendi muutmisõigustest väljaspool. Kõrvalkaust üksi ei taga kaitset. Kui mõni proov ebaõnnestub, kirjuta üles tehtud sammud, ootus ja nähtud tulemus.

4. Parandus: kirjelda täpselt üht viga

Seos õpikuga: peatükid 5 ja 8. Veakirjeldus ei anna iseenesest muutmisluba; parandus ja korduskontroll kuuluvad eraldi piiritletud sammu.

Vea leidmist saad harjutada **paberil**: kujutle, et lisasid `telk` ja `lamp` ning märkisid ainult `lamp`. **Ootus:** märgitud on üks rida. **Ettevalmistatud vea näites:** märgitakse mõlemad. Võrdle allolevat pilti ja sõnasta, mida agent peaks parandama; ära arva koodi põhjust. Arvutis saad sama viga proovida digitaalse vigase näidisfailiga. Ilma selle failita jääb tegelik agendiparandus tegemata; pelk paberiharjutus ei ole mudeli või brauseri katse.

☒ `telk`

☒ `lamp`

Ese lisatud: `lamp`. Proovisisu ei säili pärast taasavamist.

Väljalõige ettevalmistatud veast: lambi märkimisel muutuvad mõlemad read

Vihje: vaata mõlemat rida, mitte ainult teadet „Ese lisatud”. Kui sinu tulemus on teistsugune, kontrolli faili nime ja asukohta ning ava õige fail uuesti.

Tegelikuks parandamiseks loo eraldi tühi töökaust ja kopeeri sinna digitaalne vigane näidis nimega `index.html`. Paberil võid teha ainult vea tagasiside, korduskontrolli ja peatamisotsuse harjutuse. Pi käivitamisel kasuta samas piiratud keskkonnas tööriistade valikut `--tools read,write,edit` varasema `--tools write` asemel: nüüd peab agent olemasolevat faili lugema ja muutma. Bash pole paranduseks vajalik. Ära anna agendile töövihiku originaalide ega toimiva taastekoopia muutmise õigust. Täida enne parandust uus sammu 2 eelarve ja anna ühe paranduskatse luba; see võib sisaldada mitut päringut.

Kasuta töövihiku lõpus trükitud lühikest **paranduse juhist** (või selle digitaalset koopiat). Seal on samad sammud, ootus ja nähtud viga; ära kirjuta põhjust koodi oletades. Pärast lubatud parandust ava fail uuesti ning korda **kogu sammu 3 tabelit**. Nii märkad ka seda, kui parandus rikkus varem töötanud lisamise.

☐ `telk`

☒ `lamp`

Ese lisatud: `lamp`. Proovisisu ei säili pärast taasavamist.

Väljalõige parandatud näidist: telk jääb märkimata, lamp on märgitud

Lahenduse tunnus: kumbki rida on eraldi märgitav ning teised proovid läbivad. Võrdluseks on parandatud näidisfail. Näidisfaili kopeerimine on taastamine, mitte sinu agendi tehtud parandus. Kui oma parandatud fail ei läbi kontrolle, peatu; lisakatse pole automaatselt lubatud.

Paberil proovi ise: mida ütled agendile pärast pilti, millel on mõlemad read märgitud? Millised kolm varasemat käitumist kontrollid pärast parandust uuesti? **Vastusesuunis:** „Lisasin telgi ja lambi. Märkisin ainult lambi. Ootasin, et telk jääb märkimata, kuid mõlemad märgiti. Paranda ainult valitud rea märkimine.” Kontrolli taas nupu ja Enteri lisamist, tühja sisendit ning ainult ühe rea märkimist; lisaks vaata sammu 3 tabelist korduva nime ja `<lamp>` juhtumit. Kui fail või kontrollitulemus puudub, peatu — pilt ei ole sinu enda paranduskatse.

5. Taastamine: lõpeta töö enne koopia tagasipanekut

Seos õpikuga: peatükid 2 ja 8. Dialoogi lõpp, aktiivse töö lõpp ja taastatud failiseis on kolm eri asja.

Peatu kohe saladuse küsimise, kõrvaliste failimuudatuste, paigalduse, kvoodi- või makseteate korral. Peatu ka aja, käivituste, päringute või väljundtokenite piiri täitumisel.

1. Pi interaktiivses terminalis katkesta ülesanne Escape-klahviga.
2. Lõpeta Pi ning kontrolli, et see ja selle käivitatud tööprotsessid on lõppenud. Ainult terminaliakna sulgemisest või käsurea naasmisest ei piisa.
3. Vaata failiseis üle. Taasta failihalduriga kaitstud toimiv koopia töökausta nimega `index.html`; kaitstud originaali ei muudeta.
4. Ava just taastatud fail brauseris ja korda sammu 3 kontrolle. Pane tulemus kirja.

Taastamise proov enne muutmist. Kontrolli kaitstud `toimiv-index.html` sammu 3 tabeliga. Tee koopia eraldi proovikausta nimega `index.html`, asenda proovikoopia `vigane-index.html` sisuga ja tuvasta sammu 4 viga. Seejärel proovi ülaltoodud peatamise, taaste ja korduskontrolli järjekorda. Kõrvalkaust samal kontol ei taga taastealuse kaitset; vajad kontrollitud muutmisõigusteta asukohta. Kui proov ebaõnnestub, ära alusta muutvat agenti.

Kui sa ei saa aktiivse töö peatumist kinnitada, ära hakka veel faile asendama. Ära lõpeta juhuslike `kill` või `pkill` käskudega teiste tööprotsesse.

Ootus: taastatud fail käitub taas nagu toimiv koopia. Märgi taastamine tehtuks alles pärast oma koopia kontrollimist. Failiseis ja kasutajaandmed on eri asjad; selle näite proovisisu taasavamisel kaob. Faili taastamine ei võta tagasi teenusele saadetud infot ega kulutatud kvooti.

6. Uus seanss: anna üle kontrollitud tööseis

Seos õpikuga: peatükid 6 ja 9. Uue seansi esimene ülesanne on kontrollitud seisu mõistmine, mitte automaatne jätkamine.

Täida PDF-i lõpus trükitud **projektimärke** (või selle digitaalne koopia) enda tegelike tulemuste järgi. Kirjuta sinna eesmärk, lubatud fail, tehtud kontrollid, vead, taastekoopia asukoht ja järgmine väike ülesanne. Ära lisa võtmeid. **Märke pole varukoopia.**

Lõpeta senine muutmistöö. Alusta uut Pi vestlust ning anna sellele PDF-i lõpus trükitud **uue seansi juh**is (või selle digitaalne koopia) koos kontrollitud märkmega. Alguses luba ainult tööseisu arutada, mitte faile muuta. Ka see mudelipäring vajab uut sammu 2 järgi täidetud ja rakendatavat eelarvet.

Ootus: agent eristab tehtut teadmata asjadest ega kuuluta vanu kontrolle uuteks tulemusteks. Vajadusel täpsusta märget enne järgmist muutmisülesannet.

Kui agent pakub järgmiseks koodi lugemist, täpsusta: „Paku kasutaja tehtav kontroll, mitte koodilugemine.” Siin hindad nähtavat käitumist, mitte ei pea koodi selgitama.

Keskkond annab mudelile konteksti; mudel ei mäleta iseseisvalt eelmist vestlust. Ajalugu võidakse valida või lühendada ja teenus võib andmeid säilitada. Uus vestlus ei taasta faile. Sama kausta saab kasutada ka teise AI-vahendiga, kuid see ei saa automaatselt vana vestlust ega uusi õigusi. Algaja kasutab üht muutvat vahendit korraga; vahendivahetus on valikuline.

Kui tulemus on teistsugune

Sümptom	Ohutu järgmine samm
Pi käsku ei leita	Ava uus terminal ja kontrolli command -v pi; kui see ei leia käsku, kontrolli valitud paigaldusviisi ning küsi lubatud abi, ära proovi pimesi sudo
Ligipääsu vastuses pole ready	Kontrolli Pi autentimisliidest; ära prindi võtit
Näha on tasuline mudel või makseteade	Peatu; ära osta ega vali automaatset varuteed
index.html puudub	Loomine pole lõpetatud; kontrolli eelarvet, ära nimeta näidist agenditulemuseks
Brauseris paistab vana tulemus	Kontrolli faili asukohta ja ava õige fail uuesti
Viga jääb pärast parandust alles	Peatu ja taasta toimiv koopia; uut katset planeeri eraldi

Enesekontroll ja kontrollitulemuste piirid

- ☐ Tean, milline fail brauseris avati ja kust see pärineb.
- ☐ Võti pole vestluses, failis ega projektimärkmes.
- ☐ Proovisin tavakasutust ja tabeli erandjuhte.
- ☐ Võrdlesin vea korral ootust tegeliku tulemusega.
- ☐ Oskan peatada aktiivse töö ja taastada toimiva koopia.
- ☐ Projektimärge kirjeldab minu tegelikku, mitte soovitud tulemust.

Mõttepaus: kas ready ja toimiva näite pilt tõendavad edukat agenditööd? **Ei.** Esimene näitab leitud ligipääsu, teine konkreetse faili käitumist. Loomist ja parandust tõendavad sinu tegelikud failid ning tehtud kontrollid.

Pildid ja näidisfailid ei asenda sinu kontrole. Jätkamine on valikuline; kui mõni oluline eeldus või tulemus on teadmata, peatu ja märgi see üles. Kohalik sünteetiline proov ei tõenda sobivust pärisandmete või avaldamise jaoks.

Kirjandus ja juhendid

Õpik: **Tehisarupõhine tarkvaraarendus ja agentide juhtimine algajatele** (2026), peatükid 1–12. Töövihik keskendub nende põhimõtete Linuxi ja TalTechi rakendusele, mitte õpiku kõikide ülesannete asendamisele.

- Pi kasutus- ja paigaldusjuhendid: <https://pi.dev>
- TalTechi kursuse lähtekeht: <https://courses.taltech.akaver.com/>
- AI-proxy: <https://ai-proxy.cm.itcollege.ee/>

Teenuste juhendid, mudelivalik ja kasutustingimused võivad muutuda. Kontrolli enne paigaldust, konto kasutamist või päringut kehtivaid juhiseid, õigusi, kvooti ja kulu. Koolitee ei asenda kõigile avatud ligipääsu.

Õpiku autor: Jaanus Pöial. Töövihiku koostamisel on kasutatud AI abi.

Paberil kasutatavad mallid

Allolev on sama sisu mis digitaalsetes mallides. Tekstimalli pikad read on trüki jaoks murtud; JSON-i ridu ei muudeta. Täida lüngad ise ning ära kirjuta kunagi siia tegelikku API-võtit. Mall ei anna mudelipäringu või failimuudatuse luba.

Pi hpc-vllm täielik models.json näide

```
{
  "$schema": "https://pi.dev/schemas/models.schema.json",
  "providers": {
    "hpc-vllm": {
      "baseUrl": "https://ai-proxy.cm.itcollege.ee/hpcvllm/v1",
      "api": "openai-completions",
      "apiKey": "$TALTECH_API_KEY",
      "compat": {
        "supportsUsageInStreaming": true,
        "maxTokensField": "max_tokens",
        "supportsDeveloperRole": false
      },
    },
    "models": [
      {
        "id": "Qwen3.8-27B",
        "reasoning": true,
        "contextWindow": 262144,
        "thinkingLevelMap": {
          "minimal": null,
          "low": "low",
          "medium": "medium",
          "high": null,
          "xhigh": "xhigh",
          "max": null
        },
        "compat": {
          "thinkingFormat": "chat-template",
          "chatTemplateKwargs": {
            "enable_thinking": { "$var": "thinking.enabled" },
            "reasoning_effort": {
              "$var": "thinking.effort",
              "omitWhenOff": true
            }
          }
        },
        "input": ["text", "image"]
      }
    ]
  }
}
```

Loomise juhised

Loo ainult selle õppekausta index.html: eestikeelne matkavarustuse loend. Alguses on loend tühi. Lisa eset nupu või Enteriga; tühja sisendit ära lisa. Iga rida peab olema eraldi märgitav ka sama nimega esemete puhul. Näita <lamp> tekstina, mitte lehe osana. Lehel peab kohe näha olema, et sisestatu kaob pärast lehe sulgemist; ära lisa salvestust.

Hoia kogu rakendus ühes failis. Ära loo teisi faile ega lisa rakendusele võrguühendusi. Ära paigalda midagi ega avalda rakendust. Kasuta ainult kokkulepitud mudeliteenust. Ära loe saladusi või teiste projektide faile. Kui töö ebaõnnestub või kokkulepitud piir täitub, peatu; ära proovi iseseisvalt uuesti. Ütle lühidalt, mida tegid ja mida peab inimene veel brauseris kontrollima. Ära nimeta kontrolli tehtuks, kui sa seda ei teinud.

TÄIDA ENNE SAATMIST (piirid peavad toimima väljaspool vestlust):
Töökaust ja ainult index.html muutmise õigus:
Kaitstud toimiv taastekoopia:
Pakkuja/mudel: hpc-vllm / Qwen3.8-27B; kontoõigus ja kvoot:
Päringute, aja, väljundi ja kulu piir; tehniline järelevalve:
Täieliku peatamise viis:
Tasulist varuteed pole. Mall üksi ei anna mudelipäringu luba.

Paranduse juhised

Paranda ainult selle töökausta index.html-is olev märkimisviga. Lähtefail on ettevalmistatud vigase näite koopia, mitte sinu varasem tulemus. Sammud: lisasin telk ja lamp, märkisin ainult lamp. Ootus: telk jääb märkimata, lamp on märgitud. Tegelik: mõlemad read muutuvad märgituks.

Tee iga rida eraldi märgitavaks ka sama nimega esemete korral. Säilita kujundus, nupu ja Enteriga lisamine, tühja sisendi käsitlemine ning <lamp> kuvamine tekstina. Ära lisa uusi funktsioone ega muuda muid faile. Ära lisa rakendusele võrguühendusi, paigalda midagi, avalda tulemust ega loe saladusi. Kasuta ainult kokkulepitud mudeliteenust. Kui töö ebaõnnestub või piir täitub, peatu; ära tee automaatset uut katset. Ütle lühidalt, mis muutus ja mida inimene peab brauseris üle kontrollima. Ära nimeta brauserikontrolli tehtuks, kui sa seda ei teinud.

TÄIDA ENNE SAATMIST (piirid peavad toimima väljaspool vestlust):
Töökaust ja ainult index.html lugemise/muutmise õigus:
Kaitstud toimiva faili taastekoopia:
Pakkuja/mudel: hpc-vllm / Qwen3.8-27B; kontoõigus ja kvoot:
Päringute, aja, väljundi ja kulu piir; tehniline järelevalve:
Täieliku peatamise viis:
Tasulist varuteed pole. Mall üksi ei anna uut mudelipäringu luba.

Projektimärkme mall

PROJEKTIMÄRGE – minu Linuxi matkaloendi harjutus

Täida enda tegelike tulemuste järgi. Märge ei ole varukoopia ega uus töölooba.

Eesmärk: kohalik matkavarustuse loend, väljamõeldud nimed, ilma salvestuseta.

Lubatud fail ja töökaust:

Väljastatud: pärisandmed, rakenduse võrguühendused, paigaldused, avaldamine, muud failid.

Mudeliteenus: hpc-vllm / Qwen3.8-27B, minu lubatud TalTechi ligipääs.

Võtit siia ei kirjutata; mudelipäring ja kohalik rakendus on eri ühendused.

Praeguse faili päritolu (vali ja täpsusta):

☐ agendi loodud fail ☐ ettevalmistatud näide ☐ taastatud toimiv koopia

☐ agendi parandatud fail; milline lähtefail:

Tegelikult tehtud kontrollid ja tulemused:

Kuupäev/brauser:

Nupuga ja Enteriga lisamine:

Ainult valitud rea märkimine, ka korduva nime korral:

Tühi/tühikutega sisend ja <lamp> tekstina:

Teadaolevad vead ja kontrollimata kohad:

Kaitstud toimiva taastekoopia asukoht:

Taastamine läbi proovitud (jah/ei, nähtud tulemus):

Pi ja selle tööprotsessid lõpetatud (jah/ei/teadmata):

Järgmine väike ülesanne:

Praegu lubatud tegevus (esvalt ainult arutelu):

Uue ülesande mudeli-, päringu-, aja-, väljundi- ja kulupiir:

Ära lisa võtmeid, parooli, isikuandmeid ega kogu vestluse koopiat.

Ära nimeta vanu kontrole uues seansis tehtuks.

Uue seansi juhised

Alustame uut vestlust. Siin on minu kontrollitud projektimärge:

[kopeeri siia enda ajakohastatud märge ilma saladusteta]

Praegu ainult arutame tööseisu. Võta lühidalt kokku, mis on tehtud ja mis on veel teadmata. Küsi puuduvaid täpsustusi ning soovita õppijale nähtava tulemuse kontrolli, mitte koodi lugemist või käsitsi muutmist. Ära väida, et tegid ise vanu kontrole. Ära loe ega muuda faile ilma uue loata. Ära eelda vana vestluse või eelarve jätkumist; uus vestlus ei taasta faile. Järgmine muutmine lepitakse eraldi kokku.

TÄIDA ENNE SAATMIST:

Pakkuja/mudel: hpc-vllm / Qwen3.8-27B; kontoõigus ja kvoot:

Päringute, aja, väljundi ja kulu piir; tehniline järelevalve:

Peatamise viis:

Tasulist varuteed pole. Mall üksi ei anna uut päringu- ega muutmisluba.

Tehisarupõhine tarkvaraarendus ja agentide juhtimine algajatele

Töövihik

Linux

Üks väike kohalik projekt.

Sina määrad eesmärgi ja piirid,

agent valib teostuse, sina kontrollid tulemust.