

SREDNJA ELEKTRO IN RAČUNALNIŠKA ŠOLA VELENJE

Trg mladosti 3, Velenje

MLADI RAZISKOVALCI ZA RAZVOJ ŠALEŠKE DOLINE

RAZISKOVALNA NALOGA

WIFI-POVEZAVA DO PROJEKTORJA

(deljenje zaslona preko povezave WiFi)

Tematsko področje: Računalništvo

Avtor:

Luka Lah, 3. letnik

Mentor:

Miran Zevnik, univ. dipl. inž.

Velenje, 2019

Raziskovalna naloga je bila opravljena na ŠC Velenje, Elektro in računalniška šola, 2019.

Mentor:

Miran Zevnik, univ. dipl. inž.

Datum predavitve: marec 2019



By:Avtori L. Lah, Miran Zevnik

KLJUČNA DOKUMENTACIJSKA INFORMACIJA

ŠD	ŠC Velenje, šolsko leto 2018/2019
KG	deljenje zaslona/oddajanje zaslona
AV	LAH Luka
SA	ZEVNIK Miran
KZ	3320 Velenje, SLO, Trg mladosti 3
ZA	ŠC Velenje, Elektro in računalniška šola, 2019
LI	2019
IN	WIFI-POVEZAVA DO PROJEKTORJA
TD	Raziskovalna naloga
OP	VIII, 31 str., 0 tab.,0 graf.,20 slik, 0 pril., 39 vir
IJ	SL
JI	sl

AI Z vsakim dnem se število naprav povečuje in s tem imamo tudi uporabniki vedno več naprav. Večina le teh ima zaslon za prikaz vsebine. Po navadi ljudje idealiziramo večji zaslon oziroma želimo prikazati vsebino na televiziji ali projektorju, saj je gledanje večjega zaslona manj stresno za naše oči. Glede na te potrebe že obstajajo tehnologije, ki naše naprave povezujejo s televizijami, projektorji ali računalniki (Smart View, Anycast...). Po navadi pa takšne tehnologije delujejo samo na napravah enakih blagovnih znamk (npr. Samsung telefon se lahko s svojo aplikacijo poveže samo s Samsung televizijo). Namen raziskovalne naloge je bil raziskati brezžične tehnologije deljenja zaslona na računalnik Raspberry Pi z operacijskim sistemom Linux, da bi odkrili čim preprostejše implementacije deljenja zaslona ter raziskali možnosti večuporabniškega sistema za deljenje zaslonov. Namen je bil tudi opisati vzpostavitev nekaterih najlažjih postopkov deljenja zaslona, ter o tem obvestiti javnost. Ugotovili pa smo, da je uporaba takšnih tehnologij večkrat zahtevna zaradi narave operacijskega sistema. Navadno projekte (Github) ustvarijo ljudje, da poskrbijo za svoje potrebe ter specifikacije računalnika, zato pa se lahko mi na drugačnih sistemih srečamo z nekaj težavami. Poleg narejenih projektov pa obstajajo tudi implementacije znotraj Linuxa (preko kabla) in javno objavljeni izdelki (Chromecast). Seveda pa si želimo, da bi podobne tehnologije Windows sistemov prenesli tudi na sisteme Linux.

KEY WORDS DOCUMENTATION

ND ŠC Velenje, 2018/2019
CX screen mirroring/screencast
AU LAH Luka
AA ZEVNIK Miran
PP 3320 Velenje, SLO, Trg mladosti 3
PB ŠC Velenje, Elektro in računalniška šola, 2019
PY 2019
TI WIFI-CONNECTION TO PROJECTOR
DT RESEARCH WORK
NO VIII, 31 pag., 0 col.,0 gra.,20 pic., 0 anne., 39 sour.
LA SL
AL sl/en

AB On daily bases the device count multiplies and with that said we users too own more and more devices. Most of the time normal devices have displays with which they show something. Usually we people prefer watching content on Television or projector, because looking at the larger display causes less stress on our eyes. Based on our needs, technologies that connect our devices with televisions, computers or projectors already exist (Smart View, Anycast...). But in many cases those technologies work only within the device brand (e.g. Samsung smartphone can through samsung app only connect to Samsung TV). The meaning of this research work was to research wireless technologies of screen sharing to Raspberry Pi computer with operating system Linux, to find the simplest implementations of screen share and to research options of multi user system for screen and file sharing. The meaning was also to describe the establishment of some of the easiest screen sharing solutions and to inform the public about it. We found out that use of this technologies is many times complicated due to nature of Linux. Usually the projects (Github) are made by people who take care of their own needs and system specifications, because of which there can be some problems. Beside the projects there are also implemented features within Linux (through cable), and published products which can be payable or free (Chromecast). Nevertheless we still wish for Windows features to be transferred to Linux systems.

SEZNAM OKRAJŠAV IN KRATIC

% – odstotek

angl. – polni pomen iz angleškega jezika

slo. – prevod v slovenščino

dipl. – diplomirani

ERŠ – Elektro in računalniška šola

HTML – angl. Hyper Text Markup Language (slo. jezik za označevanje nadbesedila)

http – angl. hypertext transfer protocole (nadbessedilni prenosni protokol)

inž. – inženir

ISO – angl. Interational Standard Organisation (Mednarodna organizacija za standardizacijo)

npr. – na primer

OS – operacijski sistem

oz. – oziroma

ŠCVelenje – Šolski center Velenje

t. i. – tako imenovani

URL – enotni naslov vira (angleško Uniform Resource Locator) je naslov spletnih strani v svetovnem spletu

wiki – Wikipedia

www – angl. world wide web (svetovni splet)

WiFi – angl. Width Fidelity

KAZALO VSEBINE

1	UVOD	1
1.1	HIPOTEZE	1
1.2	IDEJA ZA IZDELAVO RAZISKOVALNE NALOGE	2
2	PREGLED STANJA TEHNIKE	2
3	MATERIAL IN METODE DELO	3
4	ZAČETEK RAZISKOVANJA	4
4.1	NAČINI DELJENJA ZASLONA	4
4.1.1	Screen mirroring ali deljenje (zrcaljenje) zaslona	4
4.1.2	Screencast ali oddajanje vsebin	5
4.2	PYTHON	5
4.3	ANDROID	8
4.4	Mobilna aplikacija za deljenje zaslona	8
4.4.1	Prenos vsebin s pomočjo Smart View na televizijo Samsung	10
4.5	WINDOWS	10
4.5.1	Uporaba aplikacije Connect za deljenje zaslona	11
4.5.2	Uporaba aplikacije Connect kot naprava za sprejemanje zaslona	11
4.5.3	TeamViewer in Google oddaljeno namizje	12
4.5.4	Digitalne učilnice za OS Windows	13
4.6	LINUX	14
4.6.1	Deljenje zaslona preko kabla	14
4.6.2	Omxiv	15
4.6.3	PiCAST	16
4.6.4	Miraclecast	17
4.7	PLAČLJIVI IZDELKI	17
4.8	EON BOX	18
5	ZAKLJUČEK IN UGOTOVITVE	19
6	ZAHVALA	21
7	VIRI IN LITERATURA	22
7.1	Viri slik	22
7.2	Viri vsebine	22
8	AVTOR RAZISKOVALNE NALOGE	24

KAZALO SLIK

Slika 1: Deljenje zaslona (screen mirroring), vir [1]	4
Slika 2: Shema delovanja oddajanja zaslona (screencast), lasten vir	5
Slika 3: Slika dela kode grafičnega uporabniškega vmesnika za zajemanje zaslona, lasten vir	6
Slika 4: Koda za povezovanje s strežnikom, lasten vir	7
Slika 5: Spletna stran (flask), lasten vir	7
Slika 6: Možnosti povezav za deljenje zaslona, lasten vir	9
Slika 7: Gumb za deljenje zaslona, lasten vir	9
Slika 8: URL povezava, ki jo dobimo za deljenje zaslona, lasten vir	10
Slika 9: Smart View, lasten vir	10
Slika 10: Funkcija Connect na Windows, lasten vir	11
Slika 11: Pod funkcijo Connect projiciranje v ta računalnik, lasten vir	12
Slika 12: Google oddaljeno namizje, lasten vir	12
Slika 13: TeamViewer, lasten vir	13
Slika 14: Predstavitev NETSupport School, vir [2]	14
Slika 15: ADB-mirroring, lasten vir	15
Slika 16: Izgled ikone za deljenje, lasten vir	16
Slika 17: Primer ne zagnanega programa PiCAST, lasten vir	17
Slika 18: Chromecast, vir [3]	18
Slika 19: EON BOX, vir [4]	18
Slika 20: Mladi raziskovalec Luka Lah, lasten vir	24

1 UVOD

Ljudje si ne predstavljamo sveta brez naprav. Prve izvedbe le teh so začele nastajati že v prvi polovici 20. stoletja. Najprej smo začeli razvijati napravem, kjer je bil v ospredju računalnik. Te izvedbe je bilo moč nadzirati in uporabljati le s pomočjo delčkov kode. Danes pa prehajamo v čas, ko je v središče vedno bolj postavljen človek, s tem pa so se začeli razvijati tudi vedno boljši uporabniški vmesniki (angl. GUI ali UI). Seveda pa je uporabniški vmesnik neuporaben, če nimamo zaslona, s katerim bi ga prikazali. Človeški razvoj je naše oči razvil v namen, da bi z njimi pridobivali informacije, in večja kot je slika, večji je razpon informacij, ki jih pridobimo. Čeprav so uporabniški vmesniki vedno naprednejši in uporabniku poenostavljajo uporabo naprave, pa si uporabniki želijo tudi večji zaslon, saj le ta poda informacije v naravnejši obliki in lajša obremenjenost na oči. Dandanes ima skoraj vsak uporabnik svoj telefon, televizijo ter računalnik. Navadno imata računalnik ter televizija večje zaslone ter sta tako primernejša za gledanje vsebine. Kaj pa, ko pridemo domov in smo predhodno delo opravljali na telefonu? Ali pa če ležimo na postelji in ne dosežemo računalnika ali daljinskega upravljalca za televizijo? Glede na te potrebe pa so se začele razvijati tehnologije, ki naš zaslon telefona oddajo na televizijo in računalnik. Te tehnologije so med nami že kar nekaj časa npr. Chromecast, miracast, Smart View... Seveda pa je večina izvedb plačljivih ali nekompatibilnih z znamko telefona.

1.1 HIPOTEZE

Pred raziskovanjem smo si zastavili naslednje hipoteze:

1. Izvedba oddajanja zaslona s sistemom Linux je cenejša od ostalih izdelkov na marketu
2. Izvedba oddajanja zaslona s sistemom Linux ima več prednosti kot slabosti.
3. Vse tehnologije ne bodo prenosljive na sistemu Linux
4. Takšno oddajanje zaslona je relativno hitro, kar pomeni manjši časovni zamik vsebine med telefonom ter sistemom Linux

1.2 IDEJA ZA IZDELAVO RAZISKOVALNE NALOGE

Ideja za raziskovalno nalogo izhaja iz vsakdanjega življenja. Na ERŠ imamo ali smo imeli 3 tehnologije deljenja vsebin ali zaslona. Ena takšna tehnologija je Chromecast, ki ne omogoča preverjanje zaščite (WPA2) brez dodatnega usmerjevalnika. Druga takšna tehnologija je Apple TV, ki podpira samo Apple izdelke. Tretja takšna tehnologija pa je aplikacija, kjer lahko na nadzornem računalniku vidimo zaslone vseh ostalih računalnikov (klientov) v nekem prostoru, in z njimi tudi upravljamo - digitalna učilnica (npr. iTALC), kar lahko potem prikažemo tudi na projektorju. Ta tehnologija ima to veliko prednost, da deluje kot večuporabniški sistem.

Zato nam je na misel prišla ideja, da bi naredili sistem namenjen za deljenje z raznih računalnikov, tablic in telefonov na projektorje ne glede na blagovno znamko naprave. Želeli smo nekako povezati večino funkcij iz vseh treh tehnologij, ter da bi bilo vse neodvisno od usmerjevalnika. Vse skupaj pa smo želeli narediti s pomočjo mini računalnika Raspberry Pi 3 ter sistemom Linux (Rasbian).

2 PREGLED STANJA TEHNIKE

Danes že obstaja veliko sistemov za deljenje zaslona, saj so tudi potrebe po njih dokaj visoke. Večina takšnih sistemov je plačljivih vendar, ponuja že narejen krmilnik, ki omogoča prenos vsebine. Eni takšnih sistemov so Chromecast, Anycast, Miracast... Na drugi strani pa obstajajo že implementacije znotraj naprave same, kot so na primer smartview ali aplikacije znotraj sistema naprave. Primeri aplikacij znotraj sistema so Vysor, screencast... Obstaja pa še tretja možnost, ki je izdelava svoje implementacije sistema. Tudi te izdelane implementacije narejene s strani uporabnikov, so večkrat dosegljive javno na spletišču Github. Seveda pa takšne implementacije ne obstajajo samo na Github, temveč tudi na drugih spletiščih.

V to raziskovalno nalogo smo se usmerili, da bi olajšali povezovanje naprav na projektor brez fizičnih povezav in popestrili izvajanje pouka s sodobnimi tehnologijami. Seveda pa nas je zamikala tudi tehnologija deljenja zaslona ter njeno delovanje.

3 MATERIAL IN METODE DELA

Za raziskovalno nalogo je uporabljeno predvsem sistemsko raziskovanje oziroma raziskovanje preko računalnika. Ves material, ki je uporabljen v raziskovalni nalogi je:

- Glavni računalnik (za raziskovanje metod ter tehnik)
- Raspberry Pi 3 (za vzpostavitev oddajanja zaslona) ter vsi potrebni pripomočki zanj (tipkovnica, miška, napajalnik, microSD kartica, ohišje...)

Glede na to da je bil namen poiskati tehniko oddajanja zaslona, ki deluje s pomočjo Raspberry Pi 3, smo uporabljali glavni računalnik za iskanje takšnih tehnik, ter Raspberry za nameščanje in preverjanje delovanja. Razlog za istočasno delovanje teh računalnikov je bil, ker je večkrat potrebna razlaga za usposabljanje sistema. Želeli pa smo hkrati usposabljati ter iskati nove tehnike deljenja.

Večkrat pa nam tehnologij ni uspelo vzpostaviti. Zato smo se zanesli na spletne vire ter videoposnetke, ki prikazujejo takšne vsebine. Glede na spletne vire pa smo se nato spoznali z delovanjem različnih tehnologij, ki nam jih ni uspelo vzpostaviti.

4 ZAČETEK RAZISKOVANJA

Ker pred začetkom raziskovalne naloge nismo imeli predhodnega znanja, smo najprej začeli raziskovati, kako sploh deliti zaslon z ostalimi. Vedeli smo samo, da telefoni omogočajo eno takšnih funkcij za deljenje zaslona. Predhodno pa smo poznali tudi izdelek Chromecast. Glede na funkcijo telefona (Smart View) ter Chromecast, smo se najprej usmerili v »screencast« oziroma oddajanje vsebin.

4.1 NAČINI DELJENJA ZASLONA

V računalniškem svetu poznamo dva načina brezžičnega deljenja zaslona. Razvila sta se primarno zaradi potreb po uporabi ter učinkovitosti izkoriščanja večjega zaslona. Prvi je tako imenovani »Screen mirroring« medtem ko se drugi imenuje »Screencast«.

4.1.1 Screen mirroring ali deljenje (zrcaljenje) zaslona

Prvi način deljenja, ki je med nami že kar nekaj časa je deljenje (zrcaljenje) zaslona. Deljenje zaslona je postopek pri katerem delimo (prenašamo) sliko svoje naprave na drugi zaslon brez prekinitve (zaslon mora biti ves čas aktiven - velika poraba energije). To pomeni, da se izvaja »screen mirroring« (angl. zrcaljenje zaslona), ki nam omogoča ogled vseh dejanj na naši napravi. Deljenje zaslona/screen mirroring, bi lahko poimenovali že vsa prenašanja zaslonov preko kabla HDMI ali VGA...

Tako ima tudi ta brezžični način možnost deljenja zaslona brez pomoči kablov. Takšen način je implementiran tudi v sistem Windows s pomočjo aplikacije Connect, ki omogoča povezavo na druge zaslone brezžično npr. svojo televizijo lahko nato spremenimo v dodaten zaslon računalnika.

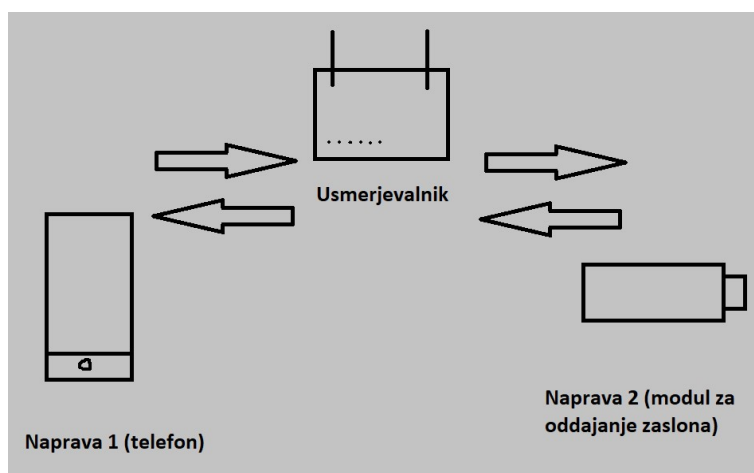
Prednost takšnega načina je, da se deljenje zaslona ne ustavi niti, ko nismo v nobeni aplikaciji (lahko delimo domači zaslon). Slabost pa je, da tukaj ni mogoče narediti večuporabniškega sistema.



Slika 1: Deljenje zaslona (screen mirroring), vir [1]

4.1.2 Screencast ali oddajanje vsebin

Drugi način oddajanje zaslona pa se je začel razvijati pred manj kot desetletjem. Ta način je screencast ali oddajanje vsebin. Oddajanje vsebin je postopek, pri katerem naprava na usmerjevalnik pošlje zahtevo za oddajanje, medtem ko naprava, ki »posluša« za takšne zahteve, to zahtevo sprejme in jo izvede. Pri izvajanju pa informacije o predvajanju vsebine naprava, ki izvaja proces, posreduje nazaj napravi, ki je oddala vsebino (slika 2). Tako imata obe napravi informacije o tem, kaj se predvaja na deljenem zaslonu. Po navadi se lahko preko tega načina oddajajo samo vsebine znotraj aplikacije v napravi. Torej oddajanje ni brez prekinitve, saj predvaja vsebino aplikacije, dokler ta ne poteče npr. pri deljenju videa iz galerije se oddajanje izključi, ko se video zaključi (lahko pa se izklopi zaslon telefona, lahko uporabljamo druge aplikacije npr. sprejem klica).



Slika 2: Shema delovanja oddajanja zaslona (screencast), lasten vir

Je pa ta tehnologija dražja, ker zahteva drugo napravo, ki bo zahteve sprejemala ter se na njih odzivala ter jih izvajala. Primeri takšnega načina deljenja zaslona je npr. Chromecast, Anycast, Miracast...

Izjema je funkcija, ki je zadnje čase implementirana v telefone Smart View, ki omogoča neprekinjeno deljenje. S pomočjo Smart View je mogoče izvajati oba načina deljenja. S tem je postajalo vse bolj jasno, da je to tehnologija, ki jo potrebujemo za izdelavo te raziskovalne naloge.

Prednosti te tehnologije so hitrost ter natančno posredovanje informacij ter večuporabniški sistem. Seveda pa ima veliko slabost, ki je seveda dodatna naprava za izvajanje tega načina.

4.2 PYTHON

Prvotna ideja je bila aplikacija v programskem jeziku Python, ki bi omogočala več uporabnikov, deljenje zaslona pa bi določal učitelj. Delili bi lahko tudi datoteke (učno gradivo).

Za začetek smo naredili aplikacijo, ki zajame posnetek zaslona ali serijo le teh, ki je omogočal modul Pillow (PIL). Za omogočanje zajemanja ter končanja zajemanja zaslona preko grafičnega uporabniškega vmesnika smo vsakemu procesu dodelili svoj thread (angl. nit) z modulom threads (slika 3).

```
start = timeit.default_timer()

root = Tk()
root.title("Screen capture")
root.configure(background="black")
root.geometry("60x100")

def makeThread1():
    t2.start()

k=False

def button1():
    a=1
    def stop():
        a=1
        global k
        k=True
        print("vrednost spremenjena")

    while(True):

        b=str(a)
        c="Slike/screenshot"+b+".jpeg"
        if k==True:
            print("k je tru")
            break
        a=a+1
        img = ImageGrab.grab()
        img.save(C, "JPEG")
        Button(root, text="Stop", command=makeThread1, fg="White", bg="Black", bd=3).c
        photo1 = PhotoImage(c)

t = threading.Thread(target=button1)

def button2():
    t.join(timeout)

t2 = threading.Thread(target=button2)

def makeThread():
    t.start()
```

Slika 3: Slika dela kode grafičnega uporabniškega vmesnika za zajemanje zaslona, lasten vir

Potem smo nadaljevali s pomočjo modula socks, kjer smo podatek pošiljali na FTP-strežnik v Filezilla (aplikacija se je prijavila v FTP ter objavila podatek, slika 4). Deljenje slik na FTP-strežnik ni bilo več tako lahko, saj zna Python na strežnik pošiljati samo Byte (objekte). Torej je bilo sliko potrebno pretvoriti v objekt Byte. Šele nato smo jo lahko poslali na strežnik. Ko pošiljamo sliko moramo strežniku seveda povedati, kakšna slika je to (jpg, png...), koliko je velikost te slike ter kakšna je resolucija.

```
import socket, pickle
import timeit
from PIL import ImageGrab, Image

start = timeit.default_timer()

server = 'localhost'
port = 800

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.connect((server, port))

user = 'USER Luka\r\n'.encode()
passwd = 'PASS LukaLah\r\n'.encode()

sock.send(user)
sock.send(passwd)
print('Connected')

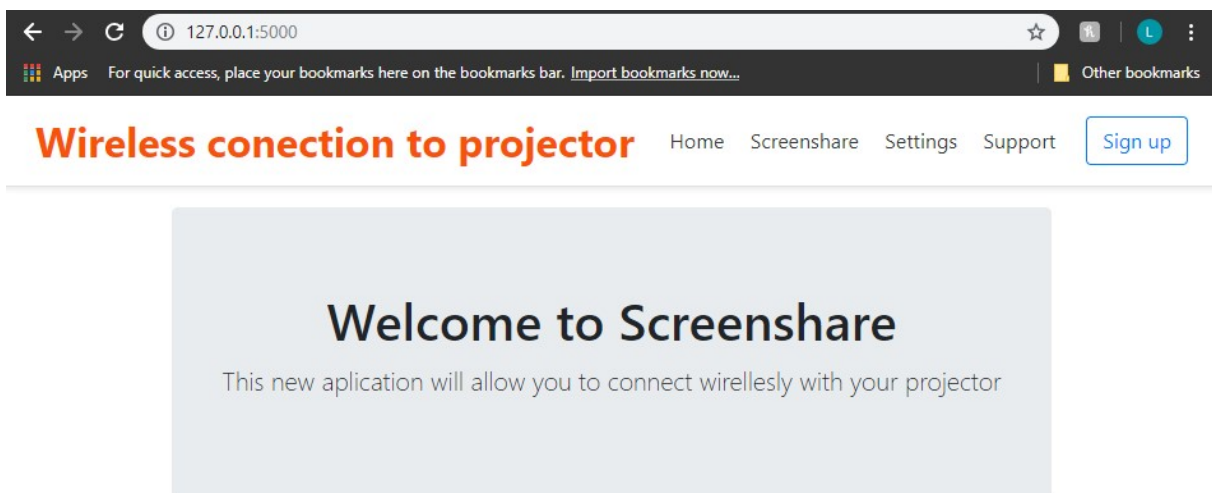
img = ImageGrab.grab()
##data_string = pickle.dumps(img)
##sock.send(data_string)

input("Click to stop connection...")
sock.close()

stop = timeit.default_timer()
print('Time: ', stop - start)
```

Slika 4: Koda za povezovanje s strežnikom, lasten vir

Vse kar je sedaj ostalo, bi bil večuporabniški sistem za deljenje podatkov, in seveda sistem, ki sprejema podatke ter jih predvaja. Želeli smo narediti spletno aplikacijo z uporabo modula Flask (slika 5). Videli pa smo, da od tukaj naprej stvar ni več tako enostavna, saj bi potrebovali ogromno znanja iz delovanja povezav. Stvar se je nato zapletla že pri ustvarjanju prijave, ki je imela težave s podatkovno bazo.



Slika 5: Spletna stran (flask), lasten vir

Slabost je bila tudi, da bi potrebovali 2 aplikaciji za napravo in sistem za sprejemanje zaslona, kar nismo želeli. Lahko bi dodali že implementirano funkcijo za deljenje zaslona pri sistemu android, vendar se je

to zaradi pomankanja znanj na področju povezav zdelo nemogoče. Problem je tudi, da večina podjetij in ljudi pojasni deljenje zaslona na preprost način, kot smo to storili mi. Zato na spletu praktično ni bilo mogoče najti natančnih podatkov za deljenje zaslona, kar je zadevo še otežilo.

Po tem smo želeli najti projekt, ki bo omogočal deljenje zaslona, saj bi tako lahko sami poskrbeli za grafični uporabniški vmesnik ali spletno stran.

4.3 ANDROID

Večina današnjih telefonov ima operacijski sistem Android (razen Apple). Na Androidu pa že obstaja vgrajena funkcija, imenovana Smart View. Kot že omenjeno, gre za oddajanje vsebine ali screencast.

Funkcija Smart View je narejena na enak način kot Windows aplikacija Connect. Razlika je samo, da android ne omogoča, da bi se nanj povezala katerakoli druga naprava. Glede na to da je operacijski sistem Android navadno nameščen na telefonih, je omogočanje drugim napravam vzpostaviti povezavo na telefon nesmiselno, saj imajo telefoni navadno manjši zaslon.

Kot raziskovalci smo si želeli tudi, da pri deljenju zaslona ne potrebujemo nameščanja drugih aplikacij. Zaradi pomankanja predznanja pa smo začeli z najlažjimi primeri deljenja zaslona.

4.4 Mobilna aplikacija za deljenje zaslona

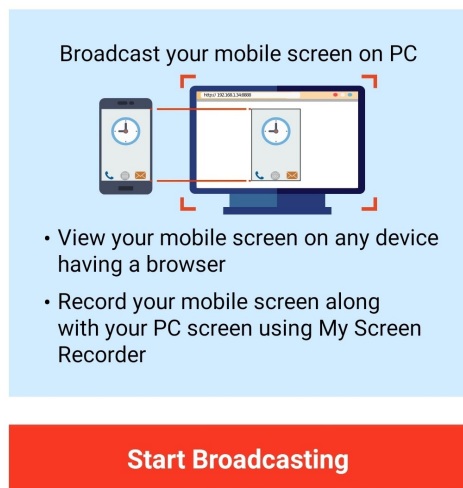
Najlažje je vzpostaviti povezavo s pomočjo mobilne aplikacije. Aplikacija za Android se imenuje Screen Cast.

Najlažja je zaradi grafičnega uporabniškega vmesnika, ki omogoča, da izberemo ali želimo zaslon deliti preko podatkovnega omrežja (mobile data), brezžičnega omrežja WiFi ali preko brezžične dostopne točke (slika 6).



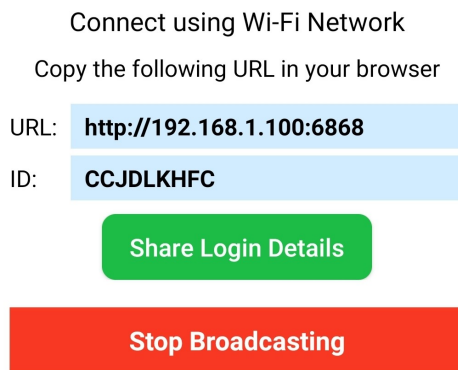
Slika 6: Možnosti povezav za deljenje zaslona, lasten vir

Torej se moramo z našim pametnim telefonom in še eno drugo napravo povezati na brezžično omrežje WiFi. Zdaj lahko na telefonu kliknemo možnost CAST (slika 7), za katero se nam pojavi URL-naslov (slika 8).



Slika 7: Gumb za deljenje zaslona, lasten vir

Vse, kar moramo narediti zdaj, je prekopirati URL-povezavo v brskalnik druge naprave. Takoj ko obiščemo to povezavo, pa lahko vidimo zaslon našega telefona. Povezavo lahko sedaj prekinemo s pomočjo drsnega elementa na telefonu. Takšna povezava je dokaj hitra, saj je oddajanje zaslona z zamikom manj kot 1 sekunde. Edina slabost pri takšni aplikaciji je potrata časa pri prepisovanju naslova URL.



Slika 8: URL povezava, ki jo dobimo za deljenje zaslona, lasten vir

4.4.1 Prenos vsebin s pomočjo Smart View na televizijo Samsung

Kot lastnik Samsung telefona ter televizije (Tizen – smart TV) smo poizkusili tudi implementirano funkcijo Samsung telefona Smart View. Tukaj je uporaba najpreprostejša. Vse, kar moramo narediti, je povezati telefon ter televizijo na enako brezžično omrežje, ter na telefonu izbrati možnost Smart View (slika 9).

Tako se nam bo odprl iskalni zavihek, ki bo našel našo televizijo. Zdaj samo še kliknemo na ime televizije in na naši televiziji se pojavi pojavno okno, ki vpraša za dovoljenje povezave. Ko smo pojavno okno potrdili, pa se vsebina, ki smo jo delili s našo televizijo, pojavi na zaslonu.

Ko smo enkrat povezani, lahko sedaj samo znotraj aplikacije izberemo možnost CAST ter delimo vsebino s televizijo. Način vzpostavitve, ki je predstavljen zgoraj, nam omogoča screen mirroring/deljenje zaslona.



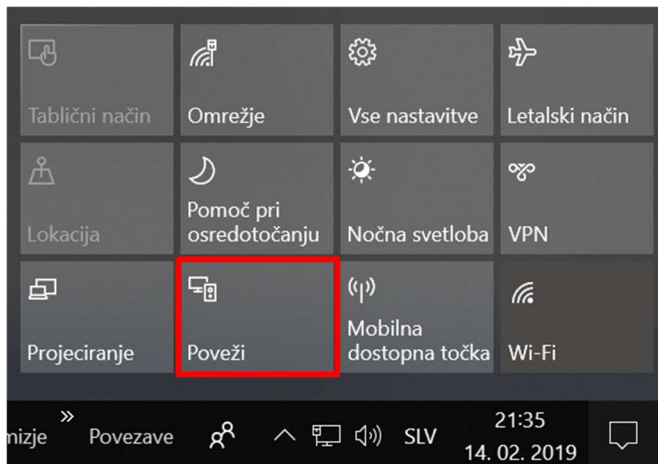
Slika 9: Smart View, lasten vir

4.5 WINDOWS

Kot že omenjeno, ima sistem Windows svojo aplikacijo, imenovano Connect. Aplikacija omogoča deljenje zaslona ali pa predstavljanje kot naprava za deljenje zaslona (kot Chromecast). Na sistemu Windows najpogosteje delimo zaslon z aplikacijo, imenovano Team viewer, ki je namenjen opravljanju računalnika na daljavo ter deljenja datotek s le-tem. Poznamo pa tudi aplikacije za šolsko uporabo, kot je npr. NetSupport School.

4.5.1 Uporaba aplikacije Connect za deljenje zaslona

Ker je aplikacija Connect zelo podobna funkciji na večini pametnih Android telefonov, je tudi vzpostavitev podobna. Na sistemu Windows preprosto odpremo ikono sporočila v skrajnem spodnjem desnem kotu (če je opravilna vrstica spodaj) ter kliknemo nanjo (slika 10). Odpre se nam nov zavihek s sporočili in funkcijami. Med funkcijami kliknemo na funkcijo Poveži/Connect, kar avtomatsko zažene iskanje naprav na istem omrežju.



Slika 10: Funkcija Connect na Windows, lasten vir

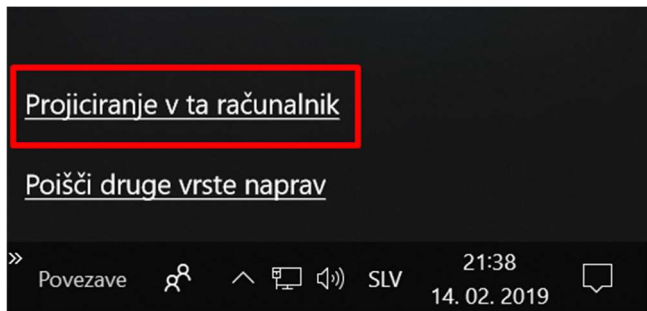
Če bo sistem našel napravo primerno za deljenje zaslona, jo bo prikazal. Po kliku na to napravo pa se skuša z njo povezati in z njo deliti zaslon. Ne samo da so naredili deljenje zaslona z drugimi napravami, ampak lahko s pomočjo aplikacije/funkcije Connect oddajamo vsebino kot drugi zaslon. To pomeni, da lahko brezžično vzpostavimo druge zaslone na računalniku sistema Windows (pri tem so lahko zasloni nastavljeni na extend).

4.5.2 Uporaba aplikacije Connect kot naprava za sprejemanje zaslona

Aplikacija Connect ali nastavev projiciranje v ta računalnik omogoča, da se Windows naprava predstavi kot naprava za predvajanje deljene vsebine (slika 11). Da omogočimo to funkcijo storimo, kot smo storili v primeru deljenja zaslona do dela, kjer kliknemo na pojavljeno napravo (na to ne kliknemo). Spodaj se nam tako pojavi možnost projiciranje v računalnik.

Po kliku na to možnost nas usmeri v nastavitve, kjer lahko to omogočimo. Ko kliknemo, da se le-ta omogoči, se nam odpre novo okno, ki ga lahko raztegnemo na celotni zaslon. V tem novem oknu se bodo sedaj prikazovali deljeni zaslone (samo 1 na enkrat seveda). S pomočjo funkcije Smart View se lahko sedaj povežemo na naš Windows računalnik.

Če naš telefon nima funkcije Smart View (Apple), pa si lahko alternativno aplikacijo naložimo in Google trgovine ali Apple trgovine. Aplikacija Connect je ena najuporabnejših funkcij za deljenje zaslona za vse operacijske sisteme.



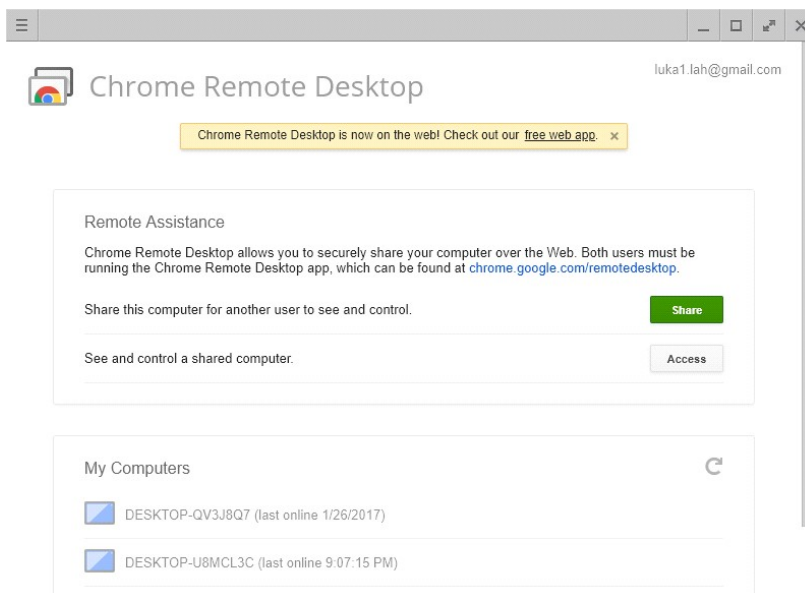
Slika 11: Pod funkcijo Connect projiciranje v ta računalnik, lasten vir

4.5.3 TeamViewer in Google oddaljeno namizje

Google oddaljeno namizje je aplikacija za oddaljen dostop do računalnika. To pomeni, da omogoča upravljanje računalnika na daljavo preko računalnika ali telefona... TeamViewer je natanko to, vendar omogoča tudi deljenje datotek...

Google oddaljeno namizje deluje kot razširitev za Google Chrome. Na trgovini razširitev si Google oddaljeno namizje dodamo, nato pa se le-ta naloži ter namesti na sistem. Čeprav je dodatek za brskalnik, potrebuje dodatno namestitev opreme za deljenje sistema (ta je avtomatska ob kliku v trgovini razširitev). Ko se le-ta namesti, moramo sedaj vpisati ime našega računalnika, PIN za dostopanje ter izbrati Google račun za Gmail, preko katerega bo potekalo deljenje.

Sedaj si npr. Na telefonu naložimo aplikacijo Google oddaljeno namizje. Na aplikaciji na telefonu se sedaj prijavimo s svojo E-pošto (slika 12) ter po izbiri računalnika vnesemo nastavljeni PIN. Tako dostopamo do našega namizja na sistemu Windows. Če bi želeli namizje upravljati z računalnikom bi to storili z razširitvijo v brskalniku.

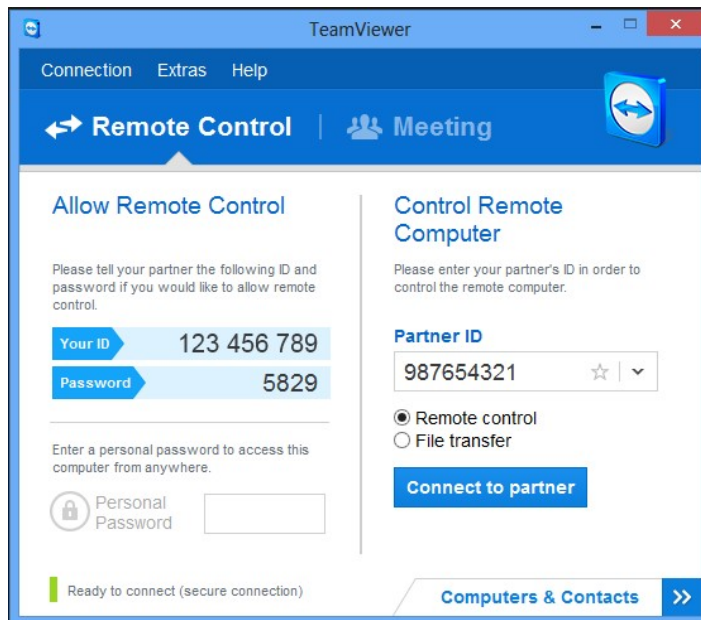


Slika 12: Google oddaljeno namizje, lasten vir

TeamViewer pa vse to stori s pomočjo PIN, ki se naključno generira ob oddajanju namizja (slika 13). Za uporabo prav tako potrebujemo aplikacijo TeamViewer na strani oddajanega namizja ter na strani,

ki dostopa do le-tega. Kot že omenjeno dobimo pri oddajanju namizja PIN, ki ga vpišemo v aplikacijo na napravi, ki želi upravljati z namizjem. Tako smo povezali Windows sistem s še eno napravo.

Ta povezava prav tako kot na Google oddaljenem namizju kot tudi na TeamViewerju, deluje na daljavo. Aplikaciji sta namenjeni za npr. uporabo domačega računalnika, ko smo na dopustu ali pa programsko servisiranje na daljavo...



Slika 13: TeamViewer, lasten vir

Slabost takšnih povezav je časovni zamik deljenja (tudi do nekaj sekund – 5 ali več). Prednost pa je interakcija s sistemom, čeprav je lahko ta tudi slabost zaradi hekerjev.

4.5.4 Digitalne učilnice za OS Windows

Digitalne učilnice omogočajo vse funkcije, ki smo si jih na začetku raziskovalne naloge zaželeli. Le-te pa imajo pomanjkljivost, da so lahko nameščene samo na mreži računalnikov z istim operacijskim sistemom (Windows). To pomeni, da ne omogoča vzpostavitve s telefonom. Še ena takšna slabost pa je, da morajo biti računalniki povezani s povezovalno strukturo drevesa.

Digitalne učilnice delujejo tako, da najprej vsakemu računalniku posebej določimo 3 vrste uporabnikov. Prva vrsta uporabnika je učenec, druga je učitelj in tretja je IT (slika 14).

Uporabniki učitelji imajo pod nadzorom svojo učilnico z uporabniki učenci. Učitelji lahko vidijo zaslone vsakega učenca. Imajo pa tudi dostop do računalnika samega in ne samo do zaslona. Učitelji lahko tako z vsemi delijo datoteke, lahko vzamejo učencu dostop do računalnika, dostop interneta...

Učenci imajo v vpogled samo svoj zaslon, funkcije ki jim jih omogočijo učitelji, ter deljene datoteke.

Uporabniki IT pa lahko odpravljajo napake, spreminjajo nastavitve, funkcije na vseh učilnicah. Tako pri potrebi po pomoči ali novi funkciji posredujejo uporabniki IT.



Slika 14: Predstavitve NETSupport School, vir [2]

4.6 LINUX

Že v povzetku in uvodu lahko vidimo, da je namen raziskati tehnologije deljenja prav tako kot vzpostaviti sistem za deljenje s pomočjo Raspberry Pi 3, ki uporablja eno od distribucij Linuxa (Raspbian). Pri raziskovanju smo naleteli na 4 tehnologije deljenja zaslona. Prva takšna je preko kabla in USB-razhroščevanja, druga preko projekta omxiv, tretja preko projektov za Screen mirroring ter četrta preko digitalne učilnice.

4.6.1 Deljenje zaslona preko kabla

Ena implementiranih funkcij Linuxa je deljenje z android napravo preko kabla t. i. ADB (angl. Android Debug Bridge) mirroring. Za delovanje te funkcije potrebujemo Linux in Android napravo.

Pri Android napravi moramo omogočiti nastavitve razvijalca. To storimo tako, da večkrat v nastavitvah kliknemo na številko gradnje (angl. Build number). Zdaj lahko v nastavitvah razvijalca omogočimo razhroščevanje USB.

Sedaj napravo priključimo na Linux sistem preko USB. V Linux terminalu smo nato napisali naslednje.

```
Sudo apt-get install android-tools-adb
```

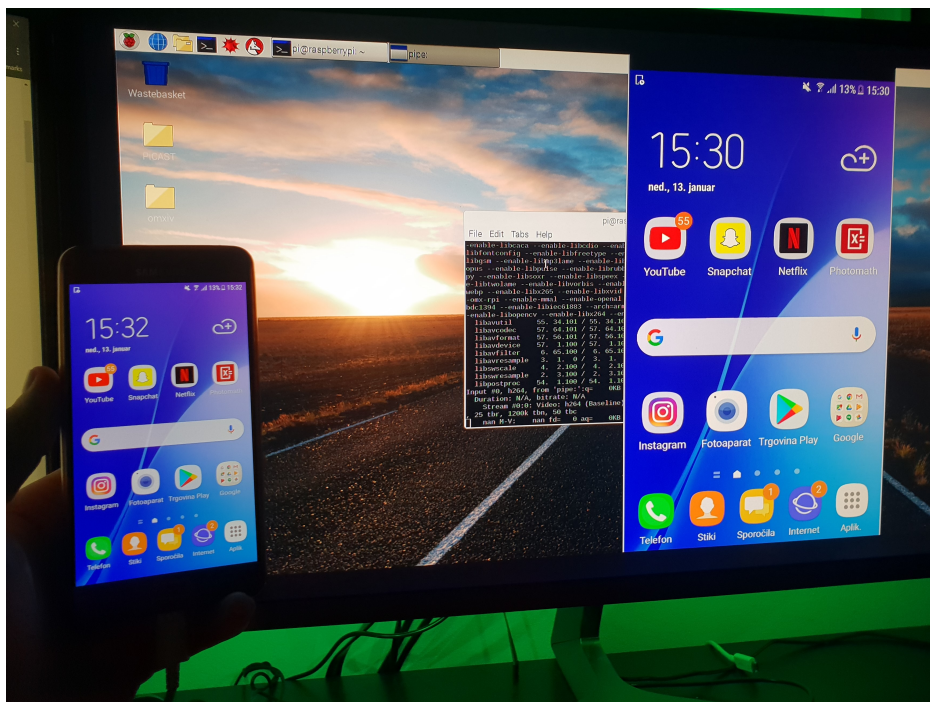
```
adb devices
```

```
Sudo apt-get install ffmpeg
```

```
Adb shell screenrecord -output-format=h264 - |ffplay -
```

S tem smo naložili ADB orodja, poizvedeli katere so priklopljene naprave, naložili ffmpeg medijski predvajalnik ter izvedli deljenje zaslona v formatu h264 (slika 15).

Takšno deljenje je na računalniku Raspberry Pi 3 skoraj neuporabno, saj ne deluje hitreje kot 1 sličico na sekundo (fps). V našem primeru se deljen zaslon ni premaknil niti po 5 minutah. To bi lahko bil problem v količini procesorske moči, ki jo ima takšen majhen računalnik. Lahko pa je bila težava v funkciji sami. Seveda pa je slabost tukaj tudi uporaba kabla.



Slika 15: ADB-mirroring, lasten vir

4.6.2 Omxiv

Omxiv (angl. OMX image viewer) je grafično pospešeni predvajalnik vsebine za Raspberry Pi 3. Deluje na principu oddajanja zaslona oziroma screen cast. Projekt je izdelal Haariger Harald. Omogoča samo posredovanje vsebine in ne tudi deljenje zaslona (screen mirroring). Delimo lahko vsebino naše naprave, kot so videoposnetki, slike, glasba in vsebino nekaterih aplikacij (youtube ...).

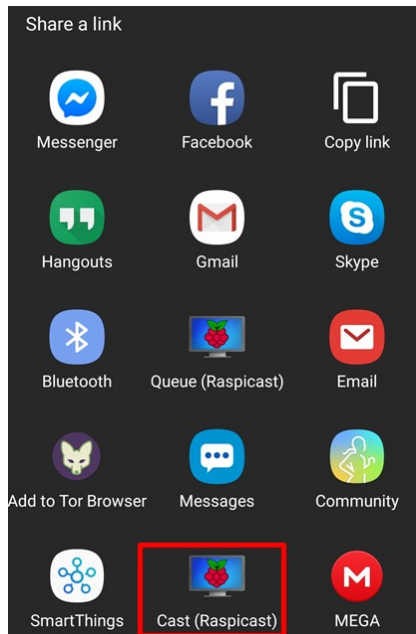
Projekt lahko naložimo na URL-naslovu »<https://github.com/HaarigerHarald/omxiv>«. To storimo tako da kopiramo povezavo dano na projektu, ter v terminal napišemo `git clone https://github.com/HaarigerHarald/omxiv`. Po tem pa se moramo postaviti v direktorij projekta. Sedaj s pomočjo naslednjih ukazov namestimo omxiv (vsa navodila so tudi na URL povezavi):

```
make ilclient
```

```
make
```

```
sudo make install
```

Zdaj moramo na napravo android samo še naložiti aplikacijo, imenovano Raspicast, in v aplikaciji nastaviti uporabniško ime, geslo ter ostale nastavitve. Vse kar moramo storiti sedaj, je odpreti aplikacijo (npr. Youtube), v tej klikniti na deljenje vsebine, ter nato vsebino deliti s pomočjo CAST(Raspicast) - Slika 16.



Slika 16: Izgled ikone za deljenje, lasten vir

Prednost takšne tehnologije je hitrost delovanja, saj je projekt narejen za minimalno procesorsko moč. Slabost pa je, da ne moremo deliti tudi domačega zaslona, temveč samo vsebino aplikacij ter medijev.

4.6.3 PiCAST

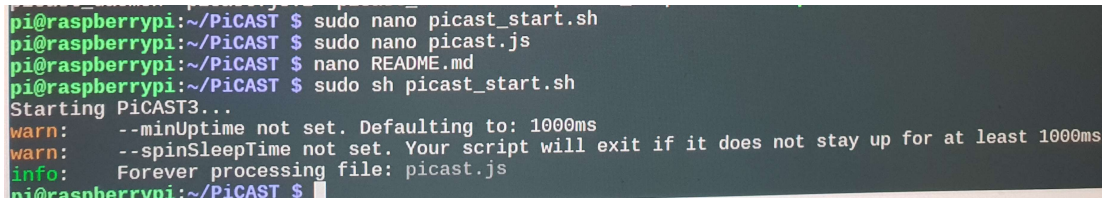
PiCAST je projekt predstavljen kot alternativa Chromecasta. Avtor projekta je Lance Seidman. Kot alternativa Chromecast naj bi podpiral oddajanje vsebin in deljenje zaslona (screencast in screen mirroring). Najprej na URL-povezavi naložimo projekt <https://github.com/lanceseidman/PiCAST> s pomočjo ukaza *git clone*.

Zdaj se postavimo v direktorij projekta in vse .sh datoteke spremenimo v izvedljive z ukazom `chmod +x datoteka.sh`. Tako bo ime datoteke.sh ob pogledu sedaj obarvano zeleno. Po tem lahko zaženemo `setup.sh`, ki bo začel z nameščanjem PiCAST. Po namestitvi se lahko projekt zažene s pomočjo ukaza `sudo sh picast_start.sh`.

Če se vse izvede tekoče, lahko sedaj odpremo okno v brskalniku z naslovom <http://ip-to-your-Pi:3000>. Na tem oknu pa lahko vidimo vse deljene vsebine.

V našem primeru pa je program javil napako (ni se zagnal). Napaka je bila v modulih js, zato smo namestili npm (paketni meneger za Java Script). S pomočjo npm smo naložili vse morebitno manjkajoče module (*npm install*). Ob ponovnem zagonu pa aplikacija znova ni delovala, zato smo poizkusili yarn (dodatna funkcija za pregledovanje ter popravljanje java script datotek). Po zagonu funkcije yarn pa aplikacija zopet ni delovala (slika 17). Vse to smo poizkusili tudi na drugih različicah PiCAST. Vse smo

poizkusili tudi ponovno naložiti in na koncu celo ponovno naložili OS za Raspberry Pi 3, ter ponovili omenjene postopke.



```
pi@raspberrypi:~/PiCAST $ sudo nano picast_start.sh
pi@raspberrypi:~/PiCAST $ sudo nano picast.js
pi@raspberrypi:~/PiCAST $ nano README.md
pi@raspberrypi:~/PiCAST $ sudo sh picast_start.sh
Starting PiCAST3...
warn: --minUptime not set. Defaulting to: 1000ms
warn: --spinSleepTime not set. Your script will exit if it does not stay up for at least 1000ms
info: Forever processing file: picast.js
pi@raspberrypi:~/PiCAST $
```

Slika 17: Primer ne zagnanega programa PiCAST, lasten vir

Projekta nam torej ni uspelo vzpostaviti.

4.6.4 Miraclecast

Miraclecast je projekt, ki naj bi bil alternativa Miracast. Avtor projekta je albfan. Projekt je dostopen na URL povezavi <https://github.com/albfan/miraclecast>. Projekt naj bi bil podoben funkciji Windowsa Connect in omogočal deljenje zaslona.

Ko je projekt enkrat naložen ga lahko namestimo s pomočjo autotools, cmake ali meson. Sami smo poizkusili vse tri možnosti, ampak so nam vse vračale napako. Autotools in cmake sta javila napake, seveda pa bi bilo popravljanje projekta preveč potratno s časom.

Nameščanje z meson pa je bilo edino, ki ni javilo napake, temveč je zahtevalo novejšo različico nekega modula. Problem pri tem je bil, da za ta modul ni obstajala takšna različica za terminal, temveč je obstajala samo Python 3.

Torej nam tudi tega projekta ni uspelo vzpostaviti.

4.7 PLAČLJIVI IZDELKI

S plačljivimi izdelki mislimo na vse izdelke, ki so tovarniško izdelani za deljenje zaslona, torej so ti izdelki namenske naprave (na napravi se upravlja samo eno delo – deljenje zaslona). Primeri takšnih naprav so Chromecast (slika 18), Miracast, Anycast...

Prednost kupljene namenske naprave je zagotovo zagotovljeno delovanje le-te. Tako omenjena prednost je tudi hitrost delovanja zaradi izvajanja enega procesa. Slabost takšnih naprav pa je še vedno nezanesljiv večuporabniški sistem in deljenje datotek (ne omogočajo).

Zaradi priložene dokumentacije, takšni sistemi vsebujejo tudi navodila za vzpostavitev ter omogočajo delovanje za večino operacijskih sistemov.



Slika 18: Chromecast, vir [3]

4.8 EON BOX

Proti koncu raziskovalne naloge smo odkrili napravo, ki omogoča večuporabniško oddajanje zaslona imenovano EON BOX. EON BOX je naprava televizijskega, internetnega ponudnika Telemach. Naprava je namenjena predvajanju vsebine televizijskih programov ponudnika Telemach (slika 19). Poleg tega pa omogoča tudi oddajanje zaslona. Napravo dobimo zraven vsebinskih paketov EON.

Funkcija oddajanja zaslona na tej napravi, za razliko od npr. Samsung televizije, posreduje podatke vsem prijavljenim uporabnikom. Nanjo lahko oddajamo zaslon s pomočjo funkcije Smart View na Androidu. Več uporabnikov se prijavi tako, da vsi vsaj enkrat prej omogočijo oddajanje zaslona z napravo EON BOX. Po tem ko smo bili že enkrat povezani, se bo naslednjič telefon na napravo EON BOX povezal sam (prižgan mora biti internet – WiFi ali Ethernet). Ko je povezanih več uporabnikov, lahko vsi vidijo, kaj se trenutno predvaja na televiziji (kaj se trenutno oddaja iz naprave).

Na EON lahko oddaja katera drugi naprava, dokler je v isti omrežni skupini (naprava na drugem usmerjevalniku se na EON ne more povezati). Tako se lahko povežemo tudi z našim hišnim računalnikom, ter katerokoli drugo napravo.

Naprava EON torej omogoča skoraj vse, kar smo si želeli vzpostaviti na začetku raziskovalne naloge. Edina stvar, ki je ne podpira, je deljenje datotek.



Slika 19: EON BOX, vir [4]

5 ZAKLJUČEK IN UGOTOVITVE

Ugotovili smo, da lahko svoje vsebine delimo na veliko različnih načinov. Deljenje pa je čisto odvisno od naših potreb. Poznamo dva načina za deljenje zaslona. Pri enem od teh delimo zgolj vsebino aplikacije (screencast), medtem ko pri drugem delimo zaslon neprekinjeno (screen mirroring). Plačljive naprave so navadno boljše od tistih, narejenih doma, saj le-te zagotavljajo delovanje, kvaliteto in podporo razvijalcev.

S pomočjo računalniškega znanja lahko danes naredimo skoraj vse, kar nam poželi srce. Eden od programskih jezikov, ki omogoča delovanje skoraj vsega, je Python. Izvedeli pa smo, da mora biti nivo znanja zelo visok za izdelavo naprednih aplikacij, napisanih z moduli za povezovanje. Za takšne aplikacije moramo sami točno in natančno poznavati vse protokole in procese v našem omrežju, ki se uporabljajo za povezovanje z in med napravami.

Ugotovili smo, da čeprav je Raspberry Pi 3 s sistemom Linux nekoliko dražji od plačljivih naprav, pa je večnamenska naprava na kateri lahko zaženemo katerikoli program/aplikacijo, ki jo ustvarimo. Seveda pa se z izdelavo lastnega modula za deljenje zaslona, tudi veliko naučimo. Naprava kot je Raspberry Pi 3 pa bi lahko poleg deljenja zaslona opravljala tudi npr. pametno inštalacijo hiše.

Ugotovili smo, da lahko zaslone delimo tudi s pomočjo aplikacij za oddaljeno namizje. Ne glede na to, da so ti programi namenjeni uporabi računalnika na daljavo, pa morajo nekako predvajati vsebino le-teh da jih lahko uporabljamo. Raba takšnih aplikacij za deljenje zaslona pa ni zaželena, saj je časovna zakasnitev zaslona lahko zelo velika (več kot 5 sekund).

Tudi Windows je na deljenje že pomislil in dodal temu primerno funkcijo Connect. Funkcija omogoča prav tako povezovanje na druge zaslone kot tudi predstavljanje kot modul za deljenje zaslona (omogoča drugim napravam povezavo do zaslona sistema Windows).

Največ pa so k deljenju prispevale digitalne učilnice, ki omogočajo veliko več, kot samo nadzor zaslonov ter deljenje le-teh. Digitalne učilnice namreč omogočajo tudi deljenje datotek, omejevanje internetne povezave ter ogleda vsebin in tudi druge funkcije, ki jih lahko ustvarimo sami oziroma oskrbovalci sistema/IT.

Največkrat pa je rešitev že pred našimi očmi. Tako s porastom naših potreb nastajajo tudi novi izdelki, ki navadno vsebujejo vse potrebne funkcionalnosti. Večkrat že znamke med svojimi izdelki naredijo funkcije za deljenje zaslona (Samsung Smart View med Samsung telefonom ter Samsung televizijo). Nekateri pa se domislijo tudi delovanje med posameznimi znamkami telefona, računalnika (EON BOX, Chromecast in drugi)...

Tako lahko prvo hipotezo zavrnilo saj je Raspberry Pi 3 nekoliko dražji od drugih izvedb, če gledaš samo napravo. Če pa upoštevamo, da morajo biti nekatere druge naprave (ChromeCast, AppleTV...) stalno priključene na internet, pa to nujno ne velja več, ker potrebujemo ponudnika za internet in po navadi še usmerjevalnik (katerega Raspberry ne potrebuje). Najcenejša izvedba deljenja zaslona bi tako bila programska. Če pa imamo doma že miniaturni računalnik (Raspberry, Compute sticks), lahko tako stroške odstranimo z lista, saj nismo zapravili denarja.

Drugo hipotezo lahko tudi zavrnilo, saj ima izvedba sistema za deljenje zaslona na sistemu Linux več slabosti kot pa prednosti. Največkrat so projekti za sistem Linux ustvarjeni za specifikacije določenega računalnika. Zato pa se lahko večkrat zgodi, da projekt na našem računalniku ne deluje. Tisti, ki imajo

veliko znanja o funkcionalnosti projekta, lahko to sami popravijo, vendar pa je to zelo časovno potratno (tako je lažje izdelati svoj projekt).

Tretjo hipotezo lahko delno potrdimo in delno zavrnemo. Potrdimo jo lahko zato, ker za večino tehnologij deljenja zaslona obstaja alternativa za sistem Linux. Zavrnemo pa jo lahko, saj ni rečeno, da bodo te tehnologije za Linux delovale tudi na našem računalniku (dveh tehnologij nam ni uspelo vzpostaviti).

Četrto in zadnjo hipotezo pa lahko tudi delno potrdimo in delno zavrnemo. Tukaj je vse odvisno od tehnologije deljenja zaslona, ki smo si jo izbrali. Npr. ko smo izbrali tehnologijo preko ADB-mirroring (kabela) je le-ta imela časovni zamik več kot 1 minuto. Zato je ta hipoteza delno zavrnjena. Delno pa jo lahko potrdimo zato, ker so tehnologije, ki nam jih je uspelo vzpostaviti, delovale relativno hitro oziroma je bil časovni zamik le-teh zanemarljiv (manj kot 1 sekunda).

6 ZAHVALA

Raziskovalna naloga ne bi bila v takšni obliki, če nam pri nastajanju le-te ne bi pomagalo veliko ljudi. Zahvala je torej namenjena naslednjim:

- mentorju Miranu Zevniku, univ. dipl. inž., za pomoč, voljo, vztrajnost, njegov prosti čas in spodbudo;
- staršem;
- Lidiji Šuster, prof., za lektoriranje;
- učiteljem ERŠ in ravnatelju Simonu Konečniku, univ. dipl. inž., za vso podporo in spodbudo;
- recenzentu raziskovalne naloge;
- komisiji Mladih raziskovalcev in koordinatorici gibanja Mladi raziskovalci Karmen Hudournik;
- vsem neomenjenim, ki so kakorkoli pomagali pri izdelavi naloge.

7 VIRI IN LITERATURA

7.1 Viri slik

Vir [1]: rizisuper. Video deljenje zaslona. <https://www.youtube.com/watch?v=WXgrt7ODHyQ> (15. 2. 2019).

Vir [2]: NetSupport. Video NetSupport School. <https://www.youtube.com/watch?v=Y7-bzwM-97w> (15. 2. 2019).

Vir [3]: Flipkart. Slika Google Chromecast 2. <https://www.flipkart.com/google-chromecast-2-media-streaming-device/p/itmeh8vyrgzymzq> (15. 2. 2019).

Vir [4]: United Group. Slika EON BOX. <https://united.group/new-eon-smart-box-launched-in-4-markets/> (15. 2. 2019).

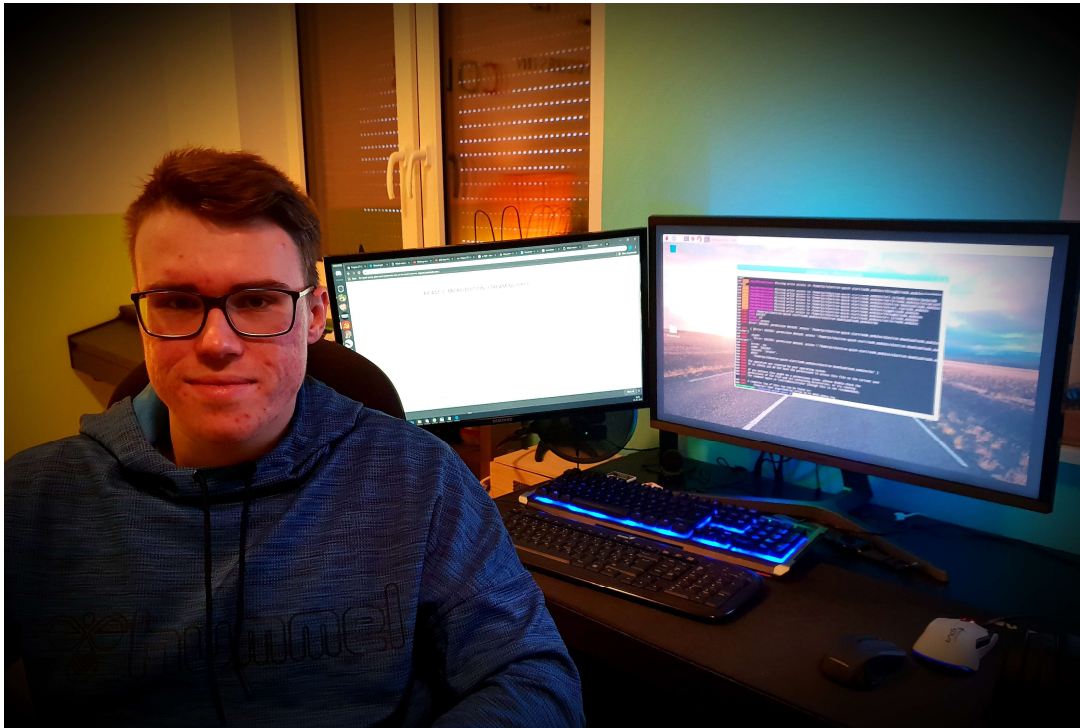
7.2 Viri vsebine

1. Daan Lenaerts. Deljenje datotek z FTP (Python). <https://daanlenaerts.com/blog/2015/08/18/python-3-upload-files-over-ftp/> (15. 2. 2019).
2. VaresanoNet. Zajemanje zaslonskih slik (Python). <http://www.varesano.net/blog/fabio/capturing%20screen%20image%20python%20and%20pil%20windows> (15. 2. 2019).
3. Python.org. FTP-klient protokol. <https://docs.python.org/3/library/ftplib.html> (15. 2. 2019).
4. Python.org. Sockets. https://drive.google.com/drive/u/0/folders/1Z1jMgE9IbZUGfAhRlqLN8Fvv_FzIDdr2 (15. 2. 2019).
5. Stackoverflow. Deljenje slike z FTP. <https://stackoverflow.com/questions/42458475/sending-image-over-sockets-only-in-python-image-can-not-be-open> (15. 2. 2019).
6. Computerworld. 10 obstoječih naprav za deljenje zaslona. <https://www.computerworld.com/article/2824189/computer-hardware/10-mirroring-devices-let-you-present-without-wires.html#slide6> (15. 2. 2019).
7. Django. Dokumentacija Django. <https://media.readthedocs.org/pdf/django-testing/latest/django-testing.pdf> (15. 2. 2019).
8. Lance Seidman. PiCAST. <https://github.com/lanceseidman/PiCAST> (15. 2. 2019).
9. Haariger Harald. OMXIV. <https://github.com/HaarigerHarald/omxiv> (15. 2. 2019).
10. Best Miracast dongle. Najbolše alternative Miracast. <https://www.bestmiracastdongle.com/miracast-alternatives/> (15. 2. 2019).
11. AlternativeTo. Alternative Miracast. <https://alternativeto.net/software/miraclecast/> (15. 2. 2019).
12. Instructables. Raspberry Pi kot alternativa Chromecast. <https://www.instructables.com/id/Raspberry-Pi-As-Chromecast-Alternative-Raspicast/> (15. 2. 2019).
13. Muo. Orodja za deljenje zaslona. <https://www.makeuseof.com/tag/12-excellent-free-screen-sharing-and-remote-access-tools-you-havent-heard-of-yet/> (15. 2. 2019).
14. AskUbuntu. Ubuntu kot Miracast pošiljatej. <https://askubuntu.com/questions/318298/ubuntu-as-miracast-sender-receiver> (15. 2. 2019).

15. Linux org. Zrcaljenje zaslona ter deljenje vsebin <https://www.linux.org/threads/screen-mirror-wireless-casting.17288/> (15. 2. 2019).
16. Raspberry Pi org. Samsung Smart View. <https://www.raspberrypi.org/forums/viewtopic.php?t=122317> (15. 2. 2019).
17. Samsung developers. Samsung Smart View SDK. <https://developer.samsung.com/smart-view> (15. 2. 2019).
18. Telemach. Naprava EON. <https://telemach.si/televizija/eon> (15. 2. 2019).
19. DevTech Solutions. Deljenje zaslona naprave Android na namizje Linux z ADB-mirroring. https://www.youtube.com/watch?v=A2IAMan_sf0 (15. 2. 2019).
20. Hackers TreeHouse. Deljenje zaslona z naprave Android na namizje Linux. <https://www.youtube.com/watch?v=Nb-5LroIV8g> (15. 2. 2019).
21. MakeTechEasier. Deljenje zaslona z naprave Android na namizje Linux. <https://www.maketecheasier.com/cast-android-screen-linux-desktop/> (15. 2. 2019).
22. How To Works. Deljenje zaslona z ADB-mirroring. <https://www.maketecheasier.com/cast-android-screen-linux-desktop/> (15. 2. 2019).
23. Virus HQ. RPI Chromecast alternativa. https://www.youtube.com/watch?v=IDQ3exG8_oM (15. 2. 2019).
24. Tech Fusion. Pretvorba računalnika Raspberry v Chromecast. <https://www.youtube.com/watch?v=yylPbtikeDA> (15. 2. 2019).
25. Being Engineers. Uporaba naprave Android kot zaslon Raspberry. <https://www.youtube.com/watch?v=sZVYzEHLcfM> (15. 2. 2019).
26. TechMe Show. PiCAST 2.0. <https://www.youtube.com/watch?v=Wwr201XHDME&t=782s> (15. 2. 2019).
27. TechMe Show. Alternativa Chromecast, PiCAST. <https://www.youtube.com/watch?v=HUgbCoORkYM&t=3s> (15. 2. 2019).
28. Stack Exchange. Povezava z Raspberry Pi preko WiFi-Direct. <https://raspberrypi.stackexchange.com/questions/45451/connect-to-raspberry-pi-3-over-wi-fi-direct> (15. 2. 2019).
29. Solder&Flux. WiFi-Direct vzpostavitev na Raspberry Pi. <http://solderandflux.com/2017/02/07/wifi-direct-style-wifi-setup-on-a-raspberry-pi/> (15. 2. 2019).
30. Reddit. Vzpostavitev WiFi-Direct na Raspberry Pi. https://www.reddit.com/r/raspberry_pi/comments/4j2bpu/how_to_setup_wifi_direct_on_a_pi3/ (15. 2. 2019).
31. Google Cast. Osnove oddajanja vsebin. https://developers.google.com/cast/docs/design_checklist/cast-basics (15. 2. 2019).
32. Quora. Kako spremeniti napravo RPi3 v napravo za zrcaljenje. <https://www.quora.com/How-do-I-change-my-Raspberry-PI-into-a-wireless-PC-mirroring-device-for-the-home-TV> (15. 2. 2019).
33. Physical Computing TV. WiFi P2P Direct. <https://www.youtube.com/watch?v=y0fNa63qbTo> (15. 2. 2019).
34. Raspberry Pi. Uporaba Pi3 kot brezžični zaslon/Miracast. <https://raspberrypi.stackexchange.com/questions/60855/using-pi-3-for-wireless-display-miracast> (15. 2. 2019).
35. Raspberry Pi. Povezava na napravo Android z WiFi-Direct. <https://raspberrypi.stackexchange.com/questions/87435/connect-to-android-device-using-wifi-direct> (15. 2. 2019).

8 AVTOR RAZISKOVALNE NALOGE

Luka Lah je dijak 3. letnika Elektro in računalniške šole (ERŠ) v Velenju. Za raziskovalno nalogo se je odločil, ker ga zanima programiranje, elektrotehnika ter nove tehnologije. Zanima ga tudi programiranje s Pythonom, Javascript ter drugimi programskimi jeziki. Zelo je aktiven tudi na športnem področju, kjer je v atletiki v državni reprezentanci. V prihodnosti se želi ukvarjati s programiranjem, avtomatizacijo procesov ter ostalih elektro ter računalniških stvari (slika 20).



Slika 20: Mladi raziskovalec Luka Lah, lasten vir