



Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Anu Eloranta

Uuden laitevarausjärjestelmän suunnittelu ja luominen käyttäen Office 365 -resurssikalentereita

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tutkinto-ohjelman nimi

Insinöörityö

10.5.2021

Tekijä Otsikko Sivumäärä Aika	Anu Eloranta Uuden laitevarausjärjestelmän suunnittelu ja luominen käyttäen Office 365 -resurssikalentereita 28 sivua 10.5.2021
Tutkinto	insinööri (AMK)
Tutkinto-ohjelma	Tieto- ja viestintätekniikka
Ammatillinen pääaine	Software engineering
Ohjaajat	Janne Salonen
Insinöörityön tarkoituksena oli suunnitella varauskalenteri käyttäen hyväksi Microsoft Office 365 -resurssikalentereita. Pää tavoitteena oli tehdä peruskoulun opettajien käyttämien kannettavien tietokoneiden varauskärryjen lainaaminen koululla helpommaksi, sekä pitää kirjaa siitä, missä kärryt ovat käytössä päivän mittaan. Tämä projekti on tehty käyttäen HTML:ää, CSS:ää ja JavaScripti kieliä, sekä Microsoft Graph -rajapintaa. Projekti on jaettu kolmeen eri osaan: infonäyttö, varaa nyt -nopea varaus sekä opettajien näyttö. Projektin aikana esiin tuli, että näitä kolmea eri käyttöliittymää ei voi yhdistää, vaan niistä tulisi tehdä kaikista omansa. Tämä johtui käytettävyydestä ja turvallisuudesta. Vain yksi, opettajien näyttö, tulisi vapaasti käytettäväksi verkon yli. Muut olisivat käytössä vain yhdeltä terminaalilta. Tässä insinöörityössä kerron, mitä palveluita käytin sekä miten suunnittelin ohjelman ja sen kaikki osat. Ohjelma on vielä sen ensimmäisessä versiossa eikä vielä produktioympäristössä. Olen myös kertonut mitkä, ominaisuudet eivät toimi halutulla tavalla ja mitä ominaisuuksia on vielä suunnitelmissa, jotta projekti saadaan käyttökuntoon.	
Avainsanat	Office 365, Microsoft Graph, Resurssikalenterit, peruskoulu, digitalisaatio, tietokoneet

Author Title	Anu Eloranta New device reservation system design and implementation using Microsoft Office 365 resource calendars
Number of Pages Date	28 pages 10 May 2021
Degree	Bachelor of Engineering
Degree Programme	Information and Communication Technology
Professional Major	Software Engineering
Instructors	Janne Salonen, Head of School
The objective of this thesis was to design a reservation calendar using Microsoft Office 365 resource calendars. The main goal was to make it easier for a teacher to reserve laptop carts that are used in upper secondary school classes. Another goal was to keep track of where the carts are. This project was done using the HTML, CSS and JavaScript languages and the Microsoft Graph API. The project was divided into three different sections, info-screen, fast reserve, and teachers' screen. During the project it became evident that there would have to be three different programs because of functionality and security and that only one of them would be freely used by the users. This thesis explains what services were used in designing the program and how certain parts were designed. The program is still in beta version and not ready to be used actively. The thesis also explains what is not working, what is planned and what other functionalities still need to be implemented for the program to reach the finish line.	
Keywords	Office 365, Microsoft Graph, Resource Calendars, upper secondary school, digitalization

Sisällys

Lyhenteet

1	Johdanto	1
2	Microsoft	2
2.1	Microsoft 365	2
2.1.1	Microsoft 365: huoneet ja laitteet	3
2.1.2	PowerShell	3
2.2	Microsoft Graph	4
2.3	Azure Active Directory	4
2.3.1	OAuth	5
2.3.2	Sovelluksen rekisteröinti Azure AD:ssa	6
3	Käytetyt tekniikat	7
3.1	HTML	7
3.2	CSS	8
3.3	JavaScript	9
3.4	API	9
4	Nykytilanne	9
4.1	Esimerkkikoulun tilanne	9
4.2	"Läppärikärryt"	10
5	Suunnittelu	12
5.1	Keskeiset ominaisuudet	14
5.2	Kolme eri ohjelmaa	14
5.2.1	Infonäyttö	14
5.2.2	Opettajan näkymä	15
5.2.3	Varaa nyt	16
6	Varauskalenteri	17
6.1	Microsoft Graph Explorer testaus	17
6.2	Infonäyttö	19
6.3	Opettajanäkymä	22

6.4	Varaa nyt -osio	23
7	Ongelmia ja kehityskohteita	25
8	Yhteenveto	26
	Lähdeluettelo	27

Lyhenteet

SPA	Single Page Application: web-sovellus tai sivusto joka, päivittää sivua dynaamisesti päivittämällä sivua uudella datalla.
HTTP	Hypertext Transfer Protocol: protokolla, joka sallii resurssien kuten HTML-dokumenttien noutamisen.
HTML	Hyper Text Markup Language: verkkosivujen luontiin kehitetty merkkikieli.
CSS	Cascading Style Sheets: käytetään antamaan HTML-sivulle tyylejä.
API	Application Programming Interface: välittäjä, joka mahdollistaa kahden sovelluksen keskustelun.
IBM	International Business Machines: teknologiayritys.
.NET	Microsoftin kehittämä avoimen lähdekoodin kehitysalusta monenlaisten sovellusten kehitykseen.
EMS	Microsoft Enterprise Mobility + Security: on älykäs liikkuvuuden hallinta- ja suojausalusta.
AD	Active Directory: Microsoft Windows -käyttäjätietokanta ja hakemisto.
ADD	Azure Active Directory: pilvipohjainen -käyttäjätietokanta ja hakemisto.
SaaS	Software as a Service, on toimintamalli, jossa ohjelmaa käytetään internetin yli. Tällöin sitä ei tarvitse asentaa tai ylläpitää paikallisesti.
OAuth	Open Authorization: avoin valtuutusprotokolla.
JSON	JavaScript Object Notation: kevyt formaatti datan lähettämiseen ja tallentamiseen.

1 Johdanto

Opinnäytetyön tarkoituksena on luoda uusi varauskalenterijärjestelmä peruskouluun opetuksessa käytettävien tietokonekärryjen varaukseen. Työn tarkoituksena on luoda yksinkertainen ja nopeasti käytettävä järjestelmä, sekä poistaa ongelmia, jotka johtuvat nykyisestä hankalista varauskalentereista.

Työ on oppimisprojekti itselleni. Aikaisemmin olen tehnyt hyvin vähän ohjelmointityötä, ja se on pääasiassa ollut HTML-ohjelmointia, joten uuden ohjelman kehittäminen on varsinainen harppaus osaamiselleni. Työtä ei varsinaisesti ole tilattu koulusta, vaan se tulee mahdollisesti käyttöön oman työni helpottamiseksi IT-vastaavana työskennellessäni.

Työssä käytetään Microsoft Graph -rajapintaa (API), jonka avulla luodaan käyttöliittymä, jolla käyttäjä voi käyttää useampaa kalenteria yhtä aikaa. Työ on Single Page Application (SPA), jossa on käytetty hyväksi HTML-, JavaScript- ja CSS-ohjelmointia. Office 365 -resurssikalenterit toimivat niin sanottuna backend-alustana eikä erillistä tietokantaa käytetä.

Opinnäytetyö koostuu neljästä osasta. Ensimmäisessä osassa käydään läpi työssä käytettäviä tekniikoita. Toisessa osassa esittelen lyhyesti koulun ja käyn läpi nykytilannetta ja sen ongelmia. Kolmanneksi esittelen työsuunnittelun ja sen toteutuksen. Viimeiseksi kerron, miten työ onnistui ja mitä jatkokehitystä se vielä tarvitsee.

Ohjelma on sen ensimmäisessä versiossa: kaikki halutut toiminnot eivät toimi aivan suunnitelman mukaan, ja sen ulkoasu ei toistaiseksi vastaa haluttua ilmettä.

2 Microsoft

Bill Gates ja Paul G. Allen perustivat Microsoftin vuonna 1975. IBM ja Microsoft solmivat 1980 sopimuksen, jossa Microsoft tekisi henkilökohtaiseen käyttöön tarkoitettuun IBM PC:hen käyttöjärjestelmän. Microsoft osti käyttöjärjestelmän toiselta yhtiöltä, muokkasi sitä ja nimesi sen MS-DOS:ksi. Joulukuussa 2020 Microsoft Windowsin markkinaosuus työasemissa on 76,56 % (Statista 2021).

Ensimmäinen versio tekstinkäsittelyohjelmasta, Word 1, julkaistiin 1983. Excel 1.0 kehitettiin alun perin Apple Macintoshille 1985 ja vasta 1987 julkaistiin Windows versio Excel 2.0. (Hall & Zachary 2020.)

2.1 Microsoft 365

Office 365 on osa Microsoft Office -tuotesarjaa. Office 365 kattaa pitkän linjan toimistotyökalut sekä pilvipohjaisen ohjelmistopakettin käyttäjäkohtaisella kuukausihinnalla.

Office 365 julkaistiin kesäkuussa 2011, alun perin yrityskäyttöön tarkoitettuna tuotteena, jatkamaan Microsoft Business Productivity Suitea. Office 2013 -julkaisun jälkeen Office 365 laajeni käsittämään erilaiset yritys ympäristöt tarjoamalla useammanlaisia lisenssiointimalleja. Myöhemmin Microsoft lisäsi lisenssit myös tavalliselle kotikäyttäjälle.

Office 365 kattaa nykyisin jokapäiväisessä työssä tutuksi tulleet toimisto-ohjelmat: Word, Excel, PowerPoint ja Outlook pilvipohjaisina sovelluksina sekä perinteisinä työpöytäsovelluksina. Käyttöympäristöä kehitetään, ja se muuttuu ja laajenee jatkuvasti, joten on erittäin vaikeaa määritellä, mikä Office 365 oikeastaan on. Lisenssimalleja on monenlaisia ja niihin kuuluu erilaisia ohjelmistoja. Opinnäytetyön kohteella on tällä hetkellä käytössä Microsoft 365 A3 -lisenssiointimalli.

Seuraavissa luvuissa kerron Microsoft 365 -sovelluksista, joita käytän tässä opinnäytetyössä.

2.1.1 Microsoft 365: huoneet ja laitteet

Huoneita ja laitteita on helppo hallinnoida Office 365:sta löytyvällä resurssien postilaatikoilla. Jos esimerkiksi kokoustilalle on luotu postilaatikko, sille muodostuu myös kalenteri. Kalenterin oikeuksien määrittäminen tapahtuu Office 365 -portaalissa, jolloin jokainen organisaation jäsen voi lisätä kokouksia eli "varauksia" resurssin kalenteriin, kun käytetään Outlook-kalenteria. Koska varauksien hallintajärjestelmän tarkoituksena on hallinnoida varauksia erillisellä sovelluksella, on kalentereihin annettava lisää oikeuksia PowerShellillä. (Microsoft 2020.)

2.1.2 PowerShell

PowerShell on Microsoftin komentotulkki, jota käytetään tehtävien automatisointiin ja koonpanon hallintaan. Toisin kuin useimmissa komentorivityökaluissa, jotka hyväksyvät ja ottavat vastaan komentoja tekstinä, PowerShell käyttää .NET-esiintymiä. PowerShellissä järjestelmän työt suoritetaan normaalisti cmdleteissä. Cmdlet on kevyt komento, joka normaalisti palauttaa .NET-esiintymän. (Microsoft 2020.)

```
PS C:\> Get-MailboxPermission
cmdlet Get-MailboxPermission at command pipeline position 1
Supply values for the following parameters:
Identity: Cart-01

Identity      User              AccessRights      IsInherited Deny
-----      -
Cart-01      NT AUTHORITY\SELF {FullAccess, ReadPermission} False      False
Cart-01      NT AUTHORITY\SELF {FullAccess, ExternalAccount, ReadPermission} False      False
```

Kuva 1. IT01-resurssikalenterin postilaatikon oikeudet

```
PS C:\> Add-MailboxFolderPermission -Identity Cart-02@munkka.fi:\Calendar -User kalenteri@munkka.fi -AccessRights Author

FolderName      User              AccessRights      SharingPermissionFlags
-----
Calendar        Kalle Kalenteri  {Author}
```

Kuva 2. IT01-resurssikalenterin oikeuksien Author lisääminen

Resurssikalentereiden oikeudet on muutettava, jotta ohjelma voi lopulta selata kaikkien kalentereiden sisältöjä.

2.2 Microsoft Graph

Microsoft Graph on rajapinta Microsoft 365:n tarjoamaan älyyn ja tietoihin. Sen avulla voidaan päästä käsiksi suureen määrään dataa, jota käyttäjä muodostaa käyttäessään Microsoft 365:ta, Windows 10- sekä Enterprise Mobility + Security -palveluita.

Microsoft Graph Rest APIa ja käyttäjäkirjastoja voidaan käyttää esimerkiksi Microsoft 365:sta OneDrivia, Teamsia, Outlookia ja monia muita palveluita. EMSin kautta saadaan pääsy esimerkiksi Azure AD:n tietoihin tai Intunesiin. Windows 10:stä voidaan käyttää aktiviteettejä tai ilmoituksia.

Voimme siis esimerkiksi lähettää käyttäjälle Windows-käyttöjärjestelmään (toast) ilmoituksia Microsoft Graphin kautta. Riittää kunhan käyttäjä on kirjautuneena jollekin koneelle Microsoft-tilillä.

Microsoft Graph avaa mahdollisuuksia järjestelmäänsä ohjelmistojen kehittäjille, mutta kaikki rajapinnan kautta tehdyt muutokset ja haut tarvitsevat aina joko adminin tai käyttäjän hyväksynnän. (Microsoft 2020.)

2.3 Azure Active Directory

Active Directory (AD), hakemisto, on tärkeä osa jokaisen organisaation verkkoympäristöä. Hybridijärjestelmässä Azuresta tulee osa organisaation infrastruktuuria. Azuren resurssien ottaminen osaksi ympäristöä mahdollistaa saumattoman autentikoinnin, hallinnoinnin ja integroinnin. Vaikka Azure Active Directory (AAD) on nimeltään Active Directory, se ei kuitenkaan ole täysiverinen hakemisto, vaan ennemminkin vain identiteetin hallintajärjestelmä. AAD on suosittu etenkin siksi, että se mahdollistaa tapauskohtaisen autentikoinnin ja valmiin federoitipalvelun. Kun ajatellaan Azure AD:ta, voidaan sitä pitää verkkopohjaisten sovellusten kuten Office 365 ja Azuren, ja satojen muiden Software-as-a-Service (SaaS) -ohjelmien autentikointipalveluna. Sallimalla autentikoinnin Azuren palvelusta ohjelmistolle kuten Twitter tai Dropbox käyttäjät voivat kirjautua omalla työpaikan tai koulun tunnuksellaan palveluun. (Microsoft 2018; Savill 2015.)

Azure AD:n avulla ohjelmistokehittäjät voivat luoda sekä yhden että useamman tenantin sovelluksia, mikä tarkoittaa, että sovellus kehitetään räätälöidysti vain yhdelle yritykselle tai yhteisölle, jolloin tenantin järjestelmänvalvoja antaa ohjelmalle oikeudet käyttää Azure AD:ta. Useamman tenantin sovelluksessa käyttäjä rekisteröi omalla AAD -tunnuksellaan sovelluksen. Jos sovellus ei ole vielä rekisteröity kyseiseen AAD:hen, sovellus rekisteröi itse itsensä. (Microsoft 2018; Savill 2015.)

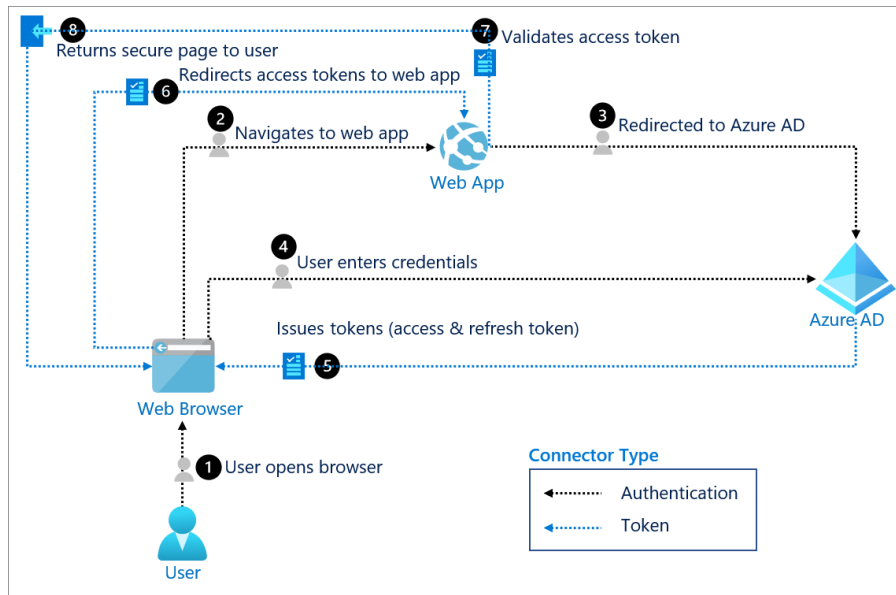
2.3.1 OAuth

OAuth on avoin standardi oikeuksien delegointiin. Sen avulla sallitaan käyttäjän pääsy rajattuun turvattuun lähteeseen. Käytämme usein tätä standardia, tiesimme siitä tai emme, esimerkiksi kirjautuessamme johonkin sovellukseen käyttämällä Googlen tai Facebookin kirjautumista.

OAuth on suunniteltu toimivan erityisesti Hypertext Transfer Protocol:n (HTTP) kanssa, sillä se erottelee asiakasohjelman roolin resurssin omistajasta. Käytännössä OAuth sallii lupaserverin, esimerkiksi Azure AD:n, muodostaa tokenin kolmannen osapuolen sovellukselle resurssin omistajan, käyttäjän, luvalla. Sovellus käyttää tokenia päästäkseen käsiksi suojattuihin resursseihin resurssipalvelimella.

Varauskalenteri pyytää tokenia Azure AD:sta, joka taas pyytää käyttäjältä luvan antaa oikeuden tiettyyn dataan Office 365:ssa, tässä tapauksessa Calendars.ReadWrite ja User.Read oikeuksiin. (Microsoft 2020.)

Kuvassa 3 on kuvattu, miten yhteistyö tokenin saamiseksi websovellukselle Azure AD:n ja käyttäjän välillä toimii.



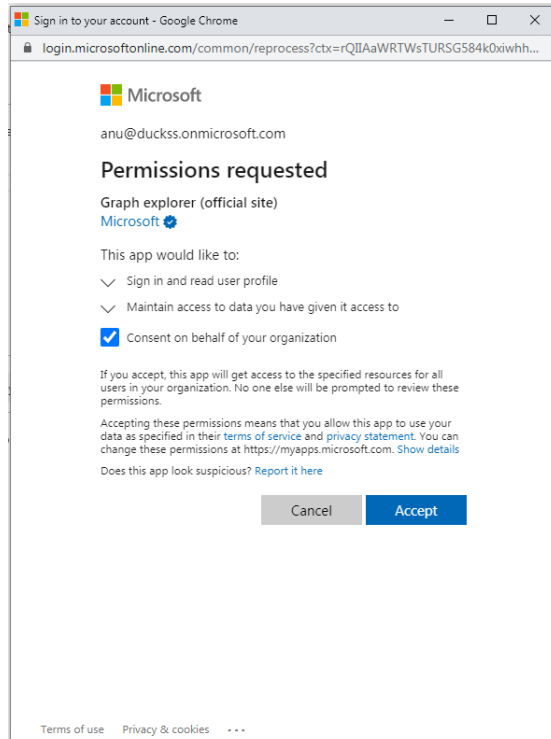
Kuva 3. Skenaario miten OAuth toimii (Microsoft 2020)

2.3.2 Sovelluksen rekisteröinti Azure AD:ssa

Kuten edellisessä kappaleessa tuli ilmi, kun sovellus pyytää tokenia Azure AD:sta, on se tätä varten rekisteröitävä Microsoftin käyttäjätietoympäristöön. Rekisteröinnissä syötetään sovelluksen tiedot Azure AD:hen ja annetaan sille oikeudet resursseihin. Resurssit voivat pyytää luvan joko käyttäjän tai järjestelmänvalvojan toimesta. Esimerkiksi käyttäjä voi antaa luvan omiin tietoihinsa, mutta vain järjestelmänvalvoja voi antaa oikeuksia kaikkien käyttäjien tietoihin.

Kun sovellus on rekisteröity, se voi pyytää lupaa käyttää käyttäjän tietoja.

Kuva 4 esittää käyttäjälle tulevaa käyttöoikeuksien hyväksymiskyselyä käyttäjän kirjautuessa sovellukseen. Graph Explorer pyytää lupaa päästä sovellusta seuraaviin tietoihin.



Kuva 4. Luvan pyytäminen Microsoft Graphia varten

3 Käytetyt tekniikat

3.1 HTML

HTML tarkoittaa HyperText Markup Languagea. HTML on yksi internetin perusrakennuspalikoista. HTML kehitettiin 1989 CERNin, joka on maailman suurin hiukkasfysiikan tutkimuslaitos Ranskan ja Sveitsin rajalla, dokumenttien uudeksi yhteiseksi kieleksi. Sen rinnalle kehitettiin yksinkertainen verkkoprotokolla http ja siitä sai alkunsa World Wide Web.

Verkkosivu luodaan käyttäen erilaisia HTML-elementtejä, joita selaimet osaavat lukea. Niiden avulla selain muuntaa koodin sivuiksi, joita käytämme. HTML erotetaan muusta tekstistä ”tägien” avulla, jotka koostuvat ”<- ja ”>” -merkeistä, kuten ”<head>”, ”<body>”, ”<div>”, ”<a>”. (W3Schools 2021.) (Docs 2021.)

```

<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>

```

Kuva 5. Yksinkertainen HTML-sivu

3.2 CSS

CSS tarkoittaa Cascading Style Sheets:iä. CSS:ää käytetään antamaan HTML-sivulle tyylejä. CSS määrittää, miten HTML-sivu näytetään selaimessa. CSS:n avulla voimme määrittää tyylin useampaan elementtiin ilman, että meidän täytyy lisätä muotoilu niihin jokaiseen erikseen. Annamme HTML-elementille luokituksen, jolle teemme CSS-tyylin. CSS-tyylisivuja kutsutaan stylesheeteiksi. (W3Schools 2021.)

```

body {
  background-color: lightblue;
}

h1 {
  color: white;
  text-align: center;
}

p {
  font-family: verdana;
  font-size: 20px;
}

```

Kuva 6. Yksinkertainen Stylesheet.

3.3 JavaScript

JavaScript on dynaaminen ohjelmointikieli. Alun perin sitä käytettiin tuomaan eloa verkkosivuihin. Nykyisin se on kaikista käytetyin kevyt ohjelmointikieli tuomaan sivuihin dynaamisuutta. JavaScript voi muuttaa HTML- ja CSS-sivua. JavaScriptiä voidaan käyttää esimerkiksi tarkastamaan, että täytetyn HTML-lomakkeen kentät on täytetty oikein. (TutorialsPoint 2021.)

3.4 API

API tulee sanoista Application Programming Interface. Se on välittäjä, joka mahdollistaa kahden sovelluksen keskustella ilman, että niiden tulee tietää, miten toinen on toteutettu. Esimerkiksi joka kerta kun käytetään sovellusta kuten Facebookia tai tarkastetaan sää-tiedotusta puhelimesta, käytetään API:a.

API antaa mahdollisuuden joustavaan ja yksinkertaiseen suunnitteluun, hallinnointiin ja käyttöön, kun kehitetään uusia ohjelmia ja työkaluja. API on yksinkertainen keino yhdistää olemassa olevan infrastruktuurin uusiin sovelluksiin pilvi-natiivin ohjelmakehityksen avulla. Se myös mahdollistaa datan jakamisen ulkopuolisten käyttäjien kanssa säilyttäen turvallisuuden ja kontrollin. (Red Hat 2021.)

4 Nykytilanne

Seuraavassa luvussa esittelen koulun lyhyesti sekä yritän avata tilannetta ja käytänteitä liittyen varausjärjestelmään.

4.1 Esimerkkikoulun tilanne

Työssäni kuvattu kohdekoulu on perustettu vuonna 1938. Siellä vaalitaan perinteitä ja arvostetaan tasokasta opetusta. Alkuperäinen koulurakennus on 1950-luvulta, siihen on

rakennettu lisää tilaa kahteen otteeseen ja sitä on ajan kuluessa peruskorjattu. Rakennuksessa on viisi kerrosta ja pinta-alaa noin 10 000 m². Koulussa toimii sekä peruskoulun yläluokat 7–9 sekä lukio: yhteensä koulussa on noin 950 oppilasta sekä 100 henkilökunnan jäsentä. Koulusta löytyy aineluokkia ja tavallisia luokkatiloja yhteensä 52 kappaletta.

Lähitulevaisuudessa kouluun on suunnitteilla laajennus, joka tarkoittaa oppilasmäärän, opetustilojen ja laitteiden määrän kasvua. Vielä ei ole päätetty, siirrytäänkö myös peruskoulun puolella henkilökohtaisiin laitteisiin vai jatkuuko yhteisten laitteiden käyttö.

Toimin koulussa IT-vastaavana. Vastaan koulun laitteiden toimivuudesta, niiden hankinnasta ja opastan niiden käytössä.

Peruskoulun koulutyö on muuttunut viime vuosina enemmän ja enemmän digitaaliseksi. Oppilaat tekevät paljon ryhmä- ja yksilötoita tunneilla tietokoneilla. Viimeisen vuoden aikana Covid-tilanne ja pakotettu etäkoulu auttoivat myös peruskoulua tekemään varsinaisen digiloikan. Kaikki opetus siirrettiin etäopetukseen varsin lyhyellä varoitusajalla keväällä 2020. Etäopetuksessa koulussa oli käytössä Microsoft Teams -ympäristö.

Kouluun paluun jälkeen olen huomannut, että suurin osa opettajista on jatkanut Teamsin käyttöä. Teams-tunnit ja kokoukset ovat vaihtuneet lähitunneiksi, mutta alustaa käytetään tiedoston jakoon ja tehtävien palautukseen varsin ahkerasti.

Toistaiseksi peruskoululaisille ei ole hankittu omia henkilökohtaisia kannettavia tietokoneita, vaan he käyttävät yhteiskäytössä olevia kannettavia omilla Office 365 -tunnuksillaan. Lukiolaisilla on omat henkilökohtaiset kannettavat tietokoneet.

4.2 ”Läppärikärret”

Koulussa on peruskoululle osoitettuja kannettavien tietokoneiden säilytyskärriä, joita opettajat varaavat ja siirtävät luokkiin. Luokkatiloja ei ole erikseen määrätty lukiolle ja peruskoululle, joten ne voivat muuttua jokaisen jakson vaihtuessa. Tämän takia emme

ole sijoittaneet tietokoneita luokkiin vaan suunnitelleet siirrettäviä tietokonekärriä. Kärriä ei ole jokaiselle peruskoulun ryhmälle, joten tieto, missä ne ovat ja kuka ne on varannut, on tärkeää. Tällä hetkellä varattavissa on kaksitoista kärriä, joissa kuissakin on 12 kannettavaa. Usein suurien luokkakokojen takia opettaja voi tarvita useamman kärryn ryhmälleen.

Koska koulurakennus on suuri ja melko sokkeloinen, on koneiden lataus ja säilytys sijoitettu yhteen kerrokseen lähelle hissiä. Näin on pyritty estämään se, että kärriä joudutaisiin siirtämään pitkiä matkoja ahtailla käytävillä välituntien aikana.



Kuva 7. Läppärikärryn "talli" latausta ja säilytystä varten



Kuva 8. Läppärikärri

Olemme käyttäneet tähän mennessä Sharepoint-kalentereita kÄrryjen varaamiseen. Kalenteriin opettaja merkitsee ajan, nimensä ja luokkatilansa. PÄÄasiassa kalenterit toimivat hyvin. Joitain ongelmia tai ennemminkin nÄrkÄstymisiÄ on sattunut.

5 Suunnittelu

Uutta jÄrjestelmÄÄ on alettu suunnitella, koska nykyinen, vaikkakin toimiva, ei kata toivottuja ominaisuuksia, ja se on aiheuttanut joitain ongelmatilanteita. Yleisin nÄistÄ on se, ettÄ varattu kÄrry ei ole "tallissa", vaan joku opettaja on ottanut sen kÄyttöönsÄ ilman varausta. YleensÄ nÄin kÄy, jos opettaja pÄÄttÄÄ kesken tunnin, tai vÄlitunnin aikana ennen tunnin alkua kÄydÄ katsomassa, onko jokin kÄrry vapaana, ja ottaa sen omaan kÄyttöönsÄ ilman varauksen tarkastamista. Voi olla, ettÄ toinen opettaja onkin varannut sen, mutta ei vielä ehtinyt hakea koneita.

TÄmän takia suunnittelussa on nyt mukana myös "Varaa nyt" -osa, joka toimisi ilman kirjautumista. TÄmÄ, kuten koko kÄrryjen lainaaminen toimii luottamusperiaatteella. Toiveena on, ettÄ jos varauksen teko olisi mahdollisimman helppoa, se tulisi myös tehtyä.

Kalentereissa ei ole minkÄÄnlaista hakuominaisuutta, jota voisi hyödyntÄÄ. TÄmÄ ei varsinaisesti ole ongelma, ehkÄ vain Ärsytys. Sharepoint-kalentereita ei voi selata yhdessÄ, vaan jokaisella kÄrryllÄ on oma kalenteri. NÄmÄ on kerÄtty Sharepoint-sivuston aloitus-sivulle putkeen. Jos yhtÄ kalenteria selaa, muut kalenterit pysyvÄt tÄssÄ viikossa. TÄmÄ on pÄivÄn selvÄÄ, mutta kun opettaja sitten selaa toisen kÄrryn kalenteria, voi muodostua tilanne, jossa jokainen kalenteri on eri viikolla.

Opettaja voi myös siirtÄÄ tai poistaa toisen tekemÄn varauksen, ja jÄrjestelmä antaa myös tehdÄ pÄÄllekkÄisiÄ varauksia. Sharepoint-kalenterin varaukset ovat vedettävissÄ paikasta toiseen: toisinaan on kÄynyt esimerkiksi niin, ettÄ vahingossa ollaan siirretty toisen varaus seuraavalle pÄivÄlle.

Varauksen tekeminen vaatii kirjautumisen Sharepointtiin. Se ei itsessÄän tuota ongelmaa, mutta hektisessÄ koulupÄivÄssÄ se tuottaa valituksia. JÄrjestelmä voi olla toisinaan erittÄin hidas ja aiheuttaa siksi ongelmia.

IT05 12 kpl HP 430, hopeakantisia koneita, opintotoimiston edessä

IT05

10. toukokuuta - 16. toukokuuta 2021

10 MAANANTAI	11 TIISTAI	12 KESKIVIIKKO	13 TORSTAI	14 PERJANTAI	15 LAUANTAI
7	+ Lisä				
8	9ABC/Sii En1	8F/ Meilo			
9	OPO/set				
10	Meilo (8f ja 8G)	9G/Sii En2	9F/Sii En2		
11					
12	9G/Sii En1	9F/Sii Ra1			
13	GE/Pug				
14		KAR 9E			
15					
16					
17					
18					

IT06 12 kpl HP 430, hopeakantisia koneita, opintotoimiston edessä

IT06

10. toukokuuta - 16. toukokuuta 2021

10 MAANANTAI	11 TIISTAI	12 KESKIVIIKKO	13 TORSTAI	14 PERJANTAI	15 LAUANTAI
7					
8	9ABC/Sii En1	8F/ Meilo			
9	Jun 8d Hy3	OPO/set			
10	Meilo (8F ja 8G)	9G/Sii En2	9F/Sii En2		
11		9c Ruu 9d En3		9c Ruu biologia	
12	9G/Sii En1	9F/Sii Ra1			
13	GE/Pug	BI/Pug			
14					
15					
16					
17					

Kuva 9. Kuva osasta läppärikärry-varaussivustoa.

5.1 Keskeiset ominaisuudet

Nykyisen järjestelmän ongelmia ja ominaisuuksia kootessani sain samalla tehtyä listan haluttavista ominaisuuksista, mitä uudessa järjestelmässä tulisi olla. Yhdeksi mielestäni tärkeimmäksi ominaisuudeksi nousi kaikkien käärykalentereiden yhtäaikainen selaaminen. Toiveena oli myös hakutoiminto, jolla käyttäjä voisi hakea aikaa, ja järjestelmä ilmoittaisi vapaat kääryt. Myös uusien varausten teko ja vanhojen muokkaaminen ja poistaminen ovat tarpeellisia toimintoja.

5.2 Kolme eri ohjelmaa

Varauskalenterin suunnittelun ja koodauksen edetessä tulin siihen tulokseen, että kaikki ohjelman eri käyttötarkoitukset eivät voi olla samassa käyttöliittymässä.

Infonäyttö tarvitsee erilaiset oikeudet kuin opettajan näyttö ja varaa nyt. Infonäytön tehtävä on vain lukea tapahtumia ilman, että käyttäjä kirjautuu järjestelmään. Tässä vaiheessa projektia ei vielä ole varsinaisesti mietitty turvallisuutta, mutta mielestäni ainakin varaa nyt -versio tulisi olla käytettävissä vain tietyltä laitteelta, koska tarkoituksena olisi, että opettajan ei erikseen tarvitsisi kirjautua järjestelmään, vaan se olisi suoraan käytössä.

Ohjelmat tarvitsevat myös varsin erilaisen käyttöliittymän. Infonäytössä käytetään hyväksi koko näyttöä ja se tehdään sopivaksi isoon näyttöruutuun. Opettaja näkymä tarvitsee joustavamman käyttöliittymän.

5.2.1 Infonäyttö

Ohjelman kehittäminen lähti tästä versiosta. Tarkoituksena on laittaa infotaulu kääryparikin läheisyyteen. Siitä käyttäjä voisi helposti ja kootusti nähdä, onko joku kääryistä vapaana, ja jos ei, koska mahdollisesti on.

Kalenteri tulisi esittää koulumaailmasta tutulla lukujärjestysmallilla, kuva 10, jolloin koko viikko on kerralla näkyvissä.

Mietin, pitäisikö kalenterinäkymän olla rullaava, niin että kuluva päivä olisi aina ylimpänä, mutta päädyin siihen tulokseen, että siitä on enemmän haittaa kuin hyötyä. Totutusti maanantai on aina ylhäällä. Olisi hyvin todennäköistä, että käyttäjä katsoisi väärää päivää, jos viikonpäivä olisi aina eri paikassa.

Varauskalenteri infonäyttö												
4.1.2021												
IT01	IT02	IT03	IT04	IT05	IT06	IT07	IT08	IT09	IT10	IT11	IT12	
1	varaus ope1					varaus ope1						
2	varaus ope1											
3				varaus ope1								
4				varaus ope1								
5						varaus ope1						
6												
7												
8												
5.1.2021												
IT01	IT02	IT03	IT04	IT05	IT06	IT07	IT08	IT09	IT10	IT11	IT12	
1	varaus ope1					varaus ope1		varaus ope1				
2	varaus ope1											
3				varaus ope1								
4				varaus ope1								
5						varaus ope1						
6												
7												
8												
6.1.2021												
IT01	IT02	IT03	IT04	IT05	IT06	IT07	IT08	IT09	IT10	IT11	IT12	
1	varaus ope1					varaus ope1						
2	varaus ope1							varaus ope1				
3				varaus ope1								
4						varaus ope1						
5								varaus ope1				
6												
7												
8												
7.1.2021												
IT01	IT02	IT03	IT04	IT05	IT06	IT07	IT08	IT09	IT10	IT11	IT12	
1	varaus ope1					varaus ope1						
2	varaus ope1											
3				varaus ope1								
4				varaus ope1								
5						varaus ope1						
6												
7												
8												
8.1.2021												
IT01	IT02	IT03	IT04	IT05	IT06	IT07	IT08	IT09	IT10	IT11	IT12	
1	varaus ope1					varaus ope1						
2	varaus ope1											
3				varaus ope1							varaus ope1	
4				varaus ope1								
5						varaus ope1						
6												
7												
8												

Kuva 10. Infonäytön layout-suunnitelma

5.2.2 Opettajan näkymä

Tämä on opettajan ja varaajan näkymä. Tässä kaikkien kääryjen varauskalenterit ovat nähtävissä kerralla ja toiminto varausten tekemiseen ja muokkaamiseen. Opettaja näkee varaukset ensimmäisessä beta-versiossa samalla lailla kuin infonäkymässä. Varausta klikkaamalla sitä voidaan muokata tai se voidaan poistaa.

Seuraavassa kehitysversiossa opettajan näkymää muokataan mobiiliystävällisemmäksi ja laitteen mukaan muotoutuvammaksi.

5.2.3 Varaa nyt

Varaa nyt -näköymästä voi varata kärryn ilman kirjautumista seuraavalle kolmelle tunnille. Jos kärryä tarvitaan pidemmän ajan päähän, on aikaa käydä varaamassa se opettajan näkymän kautta. Kolmen tunnin varausaika tulee siitä, että peruskoulussa on kolmen tunnin ainekokonaisuuksia.

Kuva 11. Varaa nyt -sovelluksen layout-suunnitelma

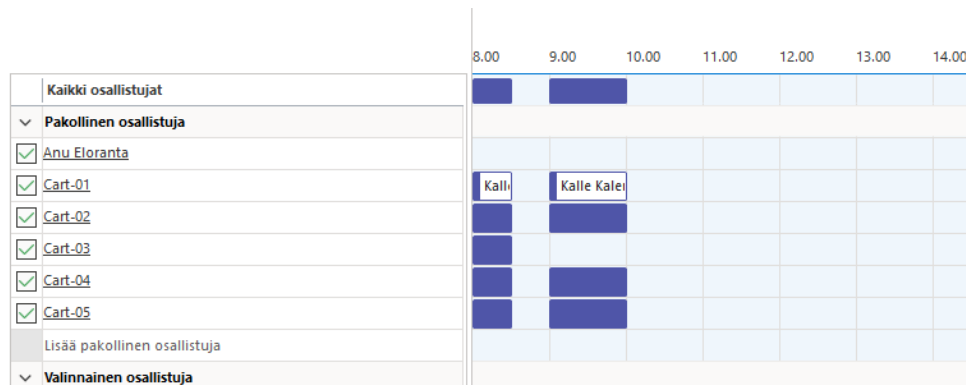
Varaa nyt -sovellus on suunniteltu käytettäväksi tabletilla. Suunnittelussa ja testauksessa olen käyttänyt vanhaa iPad 3:sta resoluutiolla 1024x768. Tarkoituksena on, että käyttäjä voisi tehdä varauksen vain muutamalla "klikkauksella".

Varaa nyt:ssä ei voi poistaa tai muuttaa varausta, ainoastaan lisätä varauksen seuraavaan kolmeen tuntiin.

6 Varauskalenteri

Työni päätarkoituksena on siis selata kalentereita ja tehdä varauksia. Tämä saattaa kuulostaa varsin yksinkertaiselta, mutta oikeasti on tarkoitus luoda ohjelma, jolla voi selata ja tehdä varauksia 12 eri kalenteriin yhtä aikaa.

Normaalisti, kun varaus tehdään Outlook-kalenterissa, se tehdään omaan kalenteriin, johon kutsutaan muita osallistujia ja he joko hyväksyvät tai hylkäävät kutsun. Työssä käytetään Microsoft Office 365:n laite- ja huonekalentereita. Kuvassa 12 on näkymä Outlook Client -ohjelmasta kalenterimerkinnän ajoitustoimintaa tehdessä. Tästä näkymästä näkee kootusti, miten jokin laitekalenteri on käytössä. On kuitenkin tehtävä kalenterimerkintä ja lisättävä siihen resurssit, ennen kuin varaukset saa näkyviin.



Kuva 12. Saatavuuden tarkistaminen Outlook-kalenterin ajoitustoiminnolla.

6.1 Microsoft Graph Explorer testaus

Aiemmin kerroin mikä Microsoft Graph on. Microsoft Graph Explorer on web-pohjainen työkalu, jolla voidaan testata erilaisia hakuja (queryjä). Seuraavissa kuvissa on muutamia havainnekuvia siitä, miten tapahtuma voidaan lisätä omaan kalenteriin ja miltä haussa saatu data näyttää.

Microsoft | Microsoft Graph Solutions Graph Explorer Getting Started Docs API Reference More All Microsoft

Graph Explorer

ae anu eloranta
anu@duckss.onmicros...

Sample queries History

cale

See more queries in the Microsoft Graph API Reference docs.

Outlook Calendar (7)

GET my events for the ...

GET all events in my cal...

GET all my calendars

POST find meeting time

POST v1.0 https://graph.microsoft.com/v1.0/me/events Run query

Request body Request headers Modify permissions Access token

```
{ "subject": "1st graph meet", "start": { "dateTime": "2020-12-22T17:34:43.681Z", "timeZone": "UTC" }, "end": { "date"
```

Created - 201 - 670ms

When you use Microsoft Graph APIs, you agree to the Microsoft APIs Terms of Use. View the Microsoft Privacy Statement

Response preview Response headers Code snippets Toolkit component Adaptive cards

```
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#users('2c1a2200-5",
  "@odata.etag": "W/\"iMOQHJ/9k6ySg6X/NB1jwAAAAAbvQ==\"",
  "id": "AAMkAGY2MmRkOGU1I TF7OGT+NDFlmN1h1hMDQ5I WT7M1j3kMTTV7M300RGAAAAAARkRn57va0JI
```

Kuva 13. Tapahtuman lisääminen omaan kalenteriin

Graph Explorer

anu eloranta
anu@duckss.onmicrosoft.com

Sample queries History

cale

See more queries in the Microsoft Graph API Reference docs.

Outlook Calendar (7)

GET my events for the next we...

GET all events in my calendar

GET all my calendars

POST find meeting time

POST schedule a meeting

POST add graph community call

GET track changes on my even...

POST v1.0 https://graph.microsoft.com/v1.0/me/events Run query

Request body Request headers Modify permissions Access token

```
{ "subject": "eka kutsu", "start": { "dateTime": "2020-12-23T17:41:10.678Z", "timeZone": "UTC" }, "end": { "dateTime": "2020-12-23T20:41:10.678Z", "timeZone": "UTC" }, "attendees": [ { "emailAddress": { "name": "IT01", "address": "IT01@duckss.onmicrosoft.com" } } ] }
```

Created - 201 - 770ms

Response preview Response headers Code snippets Toolkit component Adaptive cards Expand Share

```
{
  "type": "required",
  "status": {
    "response": "none",
    "time": "0001-01-01T00:00:00Z"
  },
  "emailAddress": {
    "name": "IT01",
    "address": "IT01@duckss.onmicrosoft.com"
  }
},
  "organizer": {
    "emailAddress": {
      "name": "anu eloranta",
      "address": "anu@duckss.onmicrosoft.com"
    }
  }
}
```

Kuva 14. Kokouskutsun lähetyksen omaan ja resurssikalenteriin



Kuva 15. Graph Explorin tuottama data haettaessa IT-02:n varauksia 19.4. klo 6-11. Kalenterissa on vain yksi varaus.

6.2 Infonäyttö

Varauskalenterin tarkoitus on, että jokainen käyttäjä voi selata resurssikalentereita vapaasti. Jokaiselle käyttäjälle on annettava vähintään lukuoikeudet resurssikalentereihin. Oikeudet annetaan PowerShell-komennolla. Vaikka Microsoftin dokumentaation mukaan reviewer -oikeuksilla käyttäjällä olisi mahdollisuus lukea tapahtumia, totesin testauksen perusteella, että author on paras vaihtoehto. Reviewer ei anna haluttuja vastauksia Graphista. (Microsoft 2020.) Kuvassa 16 on listattu kalentereiden käyttäjäroolit.

The roles that are available, along with the permissions that they assign, are described in the following list:

- Author:CreateItems, DeleteOwnedItems, EditOwnedItems, FolderVisible, ReadItems
- Contributor:CreateItems, FolderVisible
- Editor:CreateItems, DeleteAllItems, DeleteOwnedItems, EditAllItems, EditOwnedItems, FolderVisible, ReadItems
- None:FolderVisible
- NonEditingAuthor:CreateItems, FolderVisible, ReadItems
- Owner:CreateItems, CreateSubfolders, DeleteAllItems, DeleteOwnedItems, EditAllItems, EditOwnedItems, FolderContact, FolderOwner, FolderVisible, ReadItems
- PublishingEditor:CreateItems, CreateSubfolders, DeleteAllItems, DeleteOwnedItems, EditAllItems, EditOwnedItems, FolderVisible, ReadItems
- PublishingAuthor:CreateItems, CreateSubfolders, DeleteOwnedItems, EditOwnedItems, FolderVisible, ReadItems
- Reviewer:FolderVisible, ReadItems

Kuva 16. Kalenterin oikeusroolit

Kuvassa 15 näkyy tapahtuman koko JSON-objekti. Keskitymme kenttiin `start.dateTime`, `end.dateTime` ja `attendees`.

`Start.dateTime` määrittää kokouksen alkamisajan. Koska kyseessä on kalenteritapahtuma, joka halutaan esittää lukujärjestysmäisesti, aloitusajan määrittäminen on tärkeää. `End.dateTime` on loppuaika. Ohjelma käyttää alkamis- ja päättymisaikaa `div`-elementin paikan määrittämiseen palkissa. Myös tapahtuman kesto määrittää, kuinka isona palkkina tapahtumaa näytetään.

Kokouksen järjestäjä on aina ensimmäinen osallistuja, ja kun käyttäjä tekee kokouksen, johon liitetään resurssi, resurssin kalenterivarukseen tulee varaajan nimi otsikoksi.

```

async function getEvents() {
  const user = JSON.parse(sessionStorage.getItem('graphUser'));

  let ianaTimeZone = getIanaFromWindows(user.mailboxSettings.timeZone);
  let startOfWeek = String(queryViikko[0])+"00:00:00.371Z";
  let endOfWeek = String(queryViikko[4])+"23:59:59.371Z";
  console.log("start: " + startOfWeek);
  console.log("end: " + endOfWeek);

  let tapahtumaArray=[];
  let q=0;
  try {
    for(k=0; k<=11;k++){
      let nro="cart-";
      if (k<9) {
        nro+="(0" + (k+1));
      }
      else { nro+="(k+1); }

      let response = await graphClient
        .api(`users/${nro}@munkka.fi/calendarview`)
        .header("Prefer", `outlook.timezone="${user.mailboxSettings.timeZone}"`)
        .query({ startDateTime: startOfWeek, endDateTime: endOfWeek })
        .select('subject,organizer,start,end,attendees')
        .orderby(['start/datetime'])
        .top(50)
        .get();

      console.log(response.value);

      for (x=0; x<response.value.length; x++){
        tapahtumaArray[q]=response.value[x];
        q++;
      }
      nro++;
    }

    updatePage(Views.calendar, tapahtumaArray);

  } catch (error) {
    updatePage(Views.error, {
      message: 'Error getting events',
      debug: error
    });
  }
}

```

Kuva 17. Infonäytön tapahtumien luku

Infonäytön tehtävänä on lukea kaikkien kalentereiden tapahtumat kyseiseltä viikolta ja esittää ne yllänäytetyssä lukujärjestysmallissa. Kuvassa 17 koodissa nähdään, miten jokainen kalenteri luetaan erikseen objekti-taulukkoon, joka sen jälkeen asettelee varaukset oikeille paikoilleen.

Olemme myös kiinnostuneet attendeees-kentästä, koska sen avulla ohjelma jaottelee, mihin kenttään varaus tulostetaan näytölle.

Tässä versiossa ohjelma hakee aina tapahtumat uudestaan, kun selain päivitetään. Tämä ei ole toimiva ratkaisu, kun näyttö on seinällä ja tarkoitus on toimia ilman käyttäjää.

Tarkoituksena on ajastaa ohjelma skannaamaan tapahtumat 10 minuutin välein, ja jos tapahtumia on enemmän tai vähemmän kuin edellisellä kerralla, päivitetään koko näyttö. Näin päivitys voi tapahtua taustalla ilman, että käyttäjä näkee sitä näytöllä.

3.5.2021

IT01	IT02	IT03	IT04	IT05	IT06	IT07	IT08	IT09
Kalle Kalendar: 08.00-08.30								
Kalle Kalendar: 10.00-10.30								

Kuva 18. Kuva infonäytön maanantaista

6.3 Opettajanäkymä

Opettajan näkymässä on eniten toiminnallisuutta. Opettajan näkymä on vielä kesken-eräinen. Siinä toimivat selaus, uuden varauksen teko ja varauksen muokkaus ongelmitta, jos käyttäjällä on käytössään tietokone, mutta esimerkiksi mobiilisti sitä ei voi käyttää. Kuvassa 19 nähdään osa selausnäköä. Kun siirrytään edelliselle tai seuraavalla viikolle, ohjelma hakee sen viikon tapahtumat. Kuvassa 19 luodaan uusi tapahtuma.

Varauskalenteri
Kalenteri
Uusi varaus

Edellinen viikko
Tämä viikko
Seuraava viikko

3.5.2021

IT01	IT02	IT03	IT04	IT05
------	------	------	------	------

Kalle Kalenteri: 08.00-08.30

Kalle Kalenteri: 10.00-10.30

4.5.2021

Kuva 19. Opettajan selausnäky

Varauskalenteri
Kalenteri
Uusi varaus

Valitse varattava kärry

IT01

aloituspvm
18
03
2021

lopetustuspvm
18
03
2021

aloitusaika:
lopetusaika:

Create
Cancel

Kuva 20. Opettajan varauskalenteri

Toisin kuin infonäytössä käytämme opettajan kalenterissa myös id-kenttää. Tätä käytetään tapahtuman päivittämiseen ja poistamiseen.

6.4 Varaa nyt -osio

Varaa nyt -ohjelman tarkoituksena on nopea varaus. Nykyisessä järjestelmässä ongelmaksi on muodostunut, että joku vie kärryn tekemättä siitä varausta.

Skenaario on, että opettaja päättää kesken tunnin tai juuri tuntia ennen, että tällä tunnilla voitaisiin käyttää kannettavia tietokoneita. Hän menee kärryjen luokse ja katsoo infonäytöstä, että IT03-kärry on vapaana. Viereisessä opintotoimistossa on iPad, johon on asennettu paikallinen http-palvelin, joka pyörittää tätä ohjelmaa. "Ville Varaaja" on kirjautuneena ohjelmaan, josta opettaja voi nopeasti valita 1) tämä kärry, 2) tämä tunti ja 3) varaa. Ville Varaaja on Varaa nyt -näköymän oma käyttäjä.

Koska tarkoituksena on, että opettajan ei tarvitse kirjautua ohjelmaan, turvallisuussyistä ohjelma ajetaan paikallisesti.

Ohjelma luo tapahtuman, jossa varattu kärry lisätään osallistujaksi. Opettajakoodi lisätään tapahtuman otsikoksi.

Alustavasti suunnitelmassa oli, että opettaja voisi käyttää kulkutunnistetta kirjautumiseen. Toistaiseksi en ole löytänyt ratkaisua, jolla se onnistuisi ilman välissä olevaa ohjelmaa.

7 Ongelmia ja kehityskohteita

Projektissa tuli vastaan muutamia ongelmatilanteita, joita olen listannut. Osa niistä johtuu siitä, että ohjelma on vielä kehitysvaiheessa eikä sitä ole testattu suuremmassa käytössä.

Yleiseksi ongelmaksi tuli joidenkin CSS-muotoilujen toimiminen html-elementeissä. Jotkin muotoilut eivät mitenkään toimineet JavaScriptin luomassa HTML:ssä.

Infonäkymän kehityskohteina on layoutin parantaminen: toisinaan jos jossain kalenterissa ei ole tapahtumaa, seuraavan tapahtuman palkit eivät pysy kohdillaan.

Yksi infonäkymän kehityskohde on automaattinen päivitys tietyn aikavälein ilman, että selainta täytyy päivittää turhaan.

Muutamia käyttäjäperäisiä ongelmia tuli myös vastaan, muun muassa jos käyttäjä yrittää lisätä tapahtuman jo menneelle ajalle, resurssikalenteri ei hyväksy sitä. Tällöin tapahtuman luominen onnistuu, mutta tapahtumaa ei tule resurssikalenteriin. Takautuvan tapahtuman teko voi tulla vastaan esimerkiksi kesken tunnin tehtävässä varauksessa. Päivämäärän ja ajan tarkastaminen on kehityskohde sekä myös se, miten Varaa nyt -ohjelma voi tehdä varauksen jo alkaneeseen tuntiin.

8 Yhteenveto

Tämän työn tarkoituksena oli kehittää entistä toimivampi varauskalenteri koulun kannettavien tietokoneiden varauskärryille. Tähän tavoitteeseen ei täysin päästy. Ohjelma ei tällaisenaan ole käyttövalmis. Suunnitelma kuitenkin valmistui.

Ongelmatilanteita projektin kanssa tuli eteen monia. Jo se, että en ole kirjoittanut HTML:ää saati JavaScriptiä useampaan vuoteen, oli suuri haaste.

Jos aloittaisin lopputyön uudestaan, tekisin paljon johdonmukaisemman erittelyn, mitä työltä haluan, mitä ominaisuuksia vaaditaan ja mitkä ovat lisäominaisuuksia. Laatisin myös heti aluksi tarkan piirustuksen sivujen asettelusta. Tämä versio ohjelmasta lähti siitä, että testasin funktionaalisuutta, ja lopuksi yritin saada sen haluttuun muotoon. Siksi vaikka nykyisessä versiossa lähes kaikki suunniteltu toimii, ei sen ulkoasu tai käytettävyys ole sitä luokkaa, mitä valmiissa ohjelmassa haluaisin niiden olevan.

Lähdeluettelo

Azure Active Directory for developers (v1.0) overview. 2018. Verkkoaineisto. Microsoft <<https://docs.microsoft.com/en-us/azure/active-directory/azuread-dev/v1-overview>> Luettu 16.4.2020.

CSS Introduction. 2021. Verkkoaineisto. W3Schools. <https://www.w3schools.com/css/css_intro.asp> Luettu 10.5.2021.

Global market share held by operating systems for desktop PCs, from January 2013 to December 2020. 2021. Verkkoaineisto. Statista. <<https://www.statista.com/statistics/218089/global-market-share-of-windows-7/>> Luettu 14.4.2021.

Hall, M. & Zachary, G. P. 2020. Microsoft Corporation. Verkkoaineisto. Britannica. <<https://www.britannica.com/topic/Microsoft-Corporation>> Luettu 14.4.2021.

HTML Introduction. 2021. Verkkoaineisto. W3Schools. <https://www.w3schools.com/html/html_intro.asp> Luettu 10.5.2021.

HTML: HyperText Markup Language. 2021. Docs, MDN Web, Verkkoaineisto. Mozilla. <<https://developer.mozilla.org/en-US/docs/Web/HTML>> Luettu 10.5.2021.

JavaScript - Overview. 2021. Verkkoaineisto. Tutorialspoint. <https://www.tutorialspoint.com/javascript/javascript_overview.htm> Luettu 10.5.2021.

OAuth 2.0 authentication with Azure Active Directory. 2020. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/auth-oauth2>> Luettu 27.4.2021.

Overview of Microsoft Graph. 2020. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/fi-fi/graph/overview>> Luettu 13.2.2020.

Room and equipment mailboxes. 2020. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/fi-fi/microsoft-365/admin/manage/room-and-equipment-mailboxes?view=o365-worldwide>> Luettu 18.2.2020.

Savill, J., 2015. Mastering Microsoft Azure Infrastructure Services. 1 toim. Indianapolis, Indiana: Sybex, a Wiley brand.

Set-MailboxFolderPermission. 2020. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/powershell/module/exchange/set-mailboxfolderpermission?view=exchange-ps>> Luettu 14.4.2021.

What is an API? 2021. Verkkoaineisto. Red Hat. <<https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces>> Luettu 10.5.2021.

What is PowerShell? 2020. Verkkoaineisto. Microsoft <<https://docs.microsoft.com/en-us/powershell/scripting/overview?view=powershell-7.1>> Luettu 26.9.2020.