

Juha Harmonen

**MAKSUPALVELUTOIMINNAN
SELVITYS MINDHIVE OY:N
PILVIARKKITEHTUURIIN**

Opinnäytetyö
Tietojenkäsittely

2020



**Kaakkois-Suomen
ammattikorkeakoulu**

Tekijä/Tekijät	Tutkinto	Aika
Juha Harmonen	Tradenomi (AMK)	Toukokuu 2020
Opinnäytetyön nimi		57 sivua 0 liitesivua
Maksupalvelutoiminnan selvitys Mindhive Oy:n pilviarkkitehtuuriin		
Toimeksiantaja		
Mindhive Oy		
Ohjaaja		
Arto Väätäinen		
Tiivistelmä		
<p>Tämän opinnäytetyön tavoitteena oli tutkia verkkomaksun toteuttamista ja toteuttaa verkkokauppasovellus maksupalvelun API-rajapinnan avulla. Tarkoituksena oli muodostaa selkeä kuva verkkomaksun toteutuksesta API-rajapinnassa. Työssä kerrotaan, mitkä roolit ovat maksupalvelulla ja maksutapojen toimijoilla. Sitten käydään läpi verkkomaksun vaiheet ja näytetään esimerkeillä, miten tietojen vaihdossa käytetään sähköisen allekirjoituksen laskentaa.</p> <p>Tutkimusongelmana oli, kuinka maksupalveluja tulisi vertailla keskenään. Tutkimusosuudessa tutustuttiin verkkomaksupalveluihin, niistä tehtyihin tutkimuksiin ja niissä käytössä oleviin maksutapoihin. Selvisi, että maksutavat ovat tärkeintä maksupalveluissa. Maksupalveluja vertailtiin ja valittiin toimijoista Mindhivelle sopivimpia. Näille toimijoille tehtiin tarkemat tutkimukset dokumenttien ja toimijoiden haastatteluiden avulla. Lisäksi API-rajapinnat tutkittiin tekemällä niillä testejä. Testitulokset esitetään kaavioissa, joista nähdään, kuinka niitä käytettäisiin verkkokauppa-sovelluksessa ja mitä tietoja lähetetään, vastaanotetaan ja tallennetaan. Tutkimustuloksista on tehty vertaileva yhteenveto maksupalveluiden eroista. Toimijan valinta suoritettiin laskemalla vertailutulosten pisteet. Tässä tutkimuksessa ei ole merkitsevää se, kuka vertailun voitti, vaan se tapa miten ja millä kriteereillä voidaan valinta suorittaa. Tätä tapaa voidaan käyttää uusille ja tässä työssä ohitetuille maksupalveluille.</p> <p>Lopuksi käydään läpi verkkokauppa-sovelluksen toteutus. Siinä kerrotaan eri tapahtumaketjut ja miten toiminnot jaetaan frontend ja backend toteutukseen. Lisäksi näytetään esimerkillä, kuinka API-rajapintaa käytetään sovelluksessa. Verkkokauppa-sovelluksessa myydään aikaan sidottuja palveluita. Ajan varaus muuttuu tilaukseksi, vasta kun maksun suoritus on vahvistettu. Backendille tehdään prosesseja, jotka siivoavat kalenterista kesken jääneet varaukset. Tämän opinnäytetyön tuloksena on syntynyt tutkimus maksupalveluista ja verkkokauppasovellus, joka on tehty tutkimuksen ohjeiden mukaan. Johtopäätöksinä esitetään parannusehdotuksia maksupalveluiden dokumentaatioihin.</p>		
Asiasanat		
verkkomaksupalvelu, maksupalvelu, verkkomaksu, verkkokauppa, sähköinen allekirjoitus		

Author (authors)	Degree	Time
Juha Harmonen	Bachelor of Business Administration	May 2020
Thesis title		57 pages
Report on payment service operations for Mindhive Oy's cloud architecture		0 pages of appendices
Commissioned by		
Mindhive Oy		
Supervisor		
Arto Väätäinen		
<p>Abstract</p> <p>The aim of this thesis was to study online payment and implement an e-commerce application. The purpose was to provide a clear picture of the implementation of the online payment in the API of the payment service. The roles of the payment service and payment methods were explained. The thesis explained the steps of online payment and how the calculation of a digital signature was used in the exchange of information.</p> <p>The research problem was how payment services should be compared with each other. Information was collected on payment services and studies about them, as well as available payment methods. Payment methods are the most important thing in payment services. Comparisons were made between payment services and the most suitable players for Mindhive were selected. A more detailed study of these services was conducted through documentaries and interviews with the representatives of the services. APIs were tested. The test results were presented in diagrams showing how they would be used in the e-commerce application and what information would be sent, received and stored. The research results were presented in a comparative summary of the differences between payment services. The selection of the operator was made by calculating the score of the comparison results. What matters was not which service won the comparison, but how the comparison was made. This method can be used for services that were not included in this comparison.</p> <p>The thesis explained how the e-commerce application and the selected payment service were connected to each other, including the different event chains and how functions were divided between front end and back end codes. The example showed how the selected API was used in the code. The e-commerce application sells time-bound services. The time reservation has different states in the e-commerce application and it only becomes an order when the payment has been confirmed. Back end processes were needed to clean up pending payments from the order calendar. The result of this work was a study of payment services and an e-commerce application that was conducted according to the guidelines of the study. One of the conclusions was how the documentation for payment services could be improved.</p>		
<p>Keywords</p> <p>payment service, e-commerce, e-commerce application, e-payment, digital signature</p>		

SISÄLLYS

1	JOHDANTO.....	5
2	VERKKOMAKSUN TOTEUTTAMINEN.....	6
2.1	Case Mindhive oy.....	6
2.2	Maksupalvelu ja maksutapa.....	7
2.3	Verkkomaksun toteutuksen periaate.....	9
2.4	Verkkomaksun tietojen lähetys.....	13
2.5	Maksutapahtuman tietojen vastaanottaminen.....	17
2.6	Maksutapahtuman tietojen tietoturvallinen käsittely.....	20
3	MAKSUPALVELUIDEN VERTAILU MINDHIVE OY:LLE.....	22
3.1	Maksuvaihtoehtojen vertailu.....	22
3.2	Hinnoittelujen vertailu.....	24
3.3	Maksupalveluiden maksutapahtumien käsittelyn sovellukset.....	27
3.4	Maksupalveluiden vertailun tulokset.....	31
4	VALITTUJEN MAKSUPALVELUIDEN ESITTELY JA TESTIT.....	32
4.1	Paytrail.....	32
4.2	Checkout.....	36
4.3	Bambora PayForm.....	39
4.4	Maksupalvelun valinta.....	42
5	VERKKOKAUPPASOVELUKSEN TOTEUTUS.....	48
5.1	Sovelluksen tapahtumaketjut.....	48
5.2	Bamboran API:n liittäminen Backend funktioihin.....	50
5.3	Palvelun varausten käsittely.....	51
6	PÄÄTÄNTÖ.....	54
	LÄHTEET.....	56

1 JOHDANTO

Tämän työn tavoitteena on tutkia, miten verkkomaksu toteutetaan ja mitä kaikkea toteutuksessa pitää huomioida. Tutkimuksen lisäksi työssä toteutetaan verkkokauppa-sovellus. Verkkomaksussa lähetetään maksupalvelun rajapintaan tiedot tehtävästä verkko-ostoksesta ja ohjataan käyttäjä valitsemaansa maksutavan palveluun. Maksun suorituksen jälkeen käyttäjä ohjataan takaisin verkkokauppaan ja kiitetään tehdystä ostoksesta. Tämä maksutapahtuman toteutus sisältää kuitenkin salausta, varmistusta ja tiedonsiirtoa taustalla, jotta varmistetaan tietojen oikeellisuus ja maksun suoritus. Salaisten avainten käsittelyä lukuun ottamatta tästä työstä rajataan tietoturva pois.

Opinnäytetyössä tutkitaan Suomen suosituimpia maksupalveluita ja kuinka ne soveltuvat Mindhive Oy:n tarpeeseen ja kriteereihin. Tehdään vertailuja niiden ominaisuuksista, ja tutkitaan tilastoja eri maksutapojen käytöstä. Vertailun perusteella valitaan muutama haluttuun käyttötarkoitukseen parhaiten sopiva maksupalvelu, tutustutaan niiden dokumentaatioihin ja testataan palveluiden API-rajapintoja.

Maksupalvelut tarjoavat API-rajapinnat verkkomaksujen käsittelyä varten. Tämä sovellusten välinen keskustelu on keskeisessä osassa tätä opinnäytetyötä. Työssä avataan rajapinnan toiminnot, mitä niiden kautta on mahdollista tehdä ja kokeillaan testitunnuksilla eri API-toimintoja. Tietojen oikeellisuus internet-maailmassa täytyy varmistaa. Tämä on toteutettu palveluissa siirrettävien tietojen sähköisellä allekirjoituksella. Se tarkoittaa lähetettävien tietojen salauslaskentaa, käyttäen hyväksi salaista avainkoodia. Sama laskenta suoritetaan sekä lähetettäessä että vastaanotettaessa ja tuloksen on oltava sama. Verkkomaksu täytyy suojata, jotta vastikkeettomia ostotapahtumia ei pääse syntymään.

Maksupalveluiden tutkimisen jälkeen valitaan yhden toimijan palvelu. Tähän palveluun toteutetaan ohjelmamoduuli, jolla saadaan API-rajapinta ja verkkokauppa-käyttöliittymä yhdistettyä toisiinsa. Verkkoo-ostoksissa ostajalla on mahdollista myöhemmin tehdä palautus eli perua kauppa osin tai kokonaan. Tämän ja paljon muuta voi toteuttaa API-rajapinnassa. Toisaalta maksupalvelut tarjoavat myös omat kauppiaspaneelit, missä nämä palautukset voi hoitaa.

Näiden ohjelmistojen avulla verkkokauppa seuraa myös maksujen suorituksia ja tekee tilastoja ostotapahtumista.

2 VERKKOMAKSUN TOTEUTTAMINEN

Verkkomaksun suoritus on meille kaikille tuttua. Verkkokaupassa määritellään ostettavat tuotteet, syötetään ostajan tiedot ja määritellään toimitustapa. Tämän jälkeen siirrytään maksutavan palveluun suorittamaan maksu. Maksun jälkeen palataan verkkokauppaan ja saadaan tieto tilauksen onnistumisesta. Tässä luvussa kerrotaan, miten verkkomaksu toteutetaan verkkokaupan näkökulmasta. Ensin kerrotaan toimijoista, jotka ovat verkkomaksun toteutuksessa eri rooleissa. Siinä selitetään kaavion avulla, miten tiedonvaihto ja rahansiirto toimii eri toimijoiden välillä. Sitten paneudutaan, miten tämä tiedonsiirto on teknisesti toteutettu, ja mitä vaiheita siinä on. Lopuksi käsitellään salaisten avainten tietoturvallista käsittelyä.

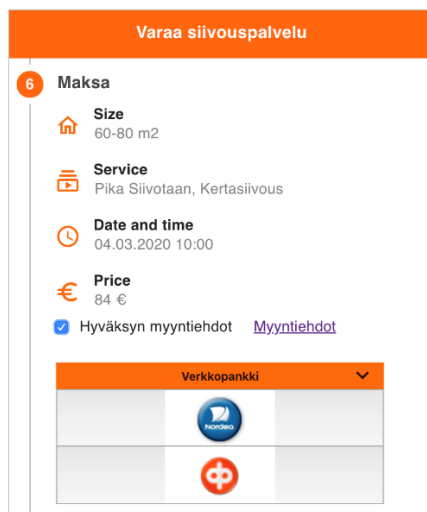
2.1 Case Mindhive oy

Mindhive on 2016 perustettu sovelluspalveluja tuottava yritys. Yrityksessä on vahvaa osaamista sovelluskehityksessä, johtamisessa, oppimisessa ja opettamisessa. Osaaminen mahdollistaa kokonaisvaltaisen palvelujen suunnittelun ja toteutuksen asiakkaan tarpeisiin. Yritys toteuttaa asiakasprojekteja, joissa rakennetaan web-sovelluksia sekä mobiilipalveluja Android- ja iOS-ympäristöihin. Mindhive Oy osallistuu myös alihankkijana web- ja mobiilisovelluskehitysprojekteihin. Yritys tuottaa yhteisöoppimisen ja tulevaisuuksien tutkimuksen vuorovaikutusta tukevia menetelmätuotteita. Yritys tekee konsultaatiota ja koulutusta erilaisiin yhteisöoppimisen menetelmämalleihin ja näitä tukeviin ratkaisuihin yrityksille ja yhteisöille.

Mindhive käyttää Googlen pilvipalveluja ja sen Serverless-ominaisuuksia. Palvelu skaalautuu automaattisesti käytön mukaan. Käyttäjälle se näkyy sovelluksen nopeutena, oli yhtäaikaisia käyttäjiä yksi tai miljoona. Automaattinen skaalaus tarkoittaa, että ei makseta turhasta kapasiteetista, eikä yrityksen tarvitse reagoida käyttöasteen muutoksiin.

Toteutettava sovellus

Toteutettava sovellus on sivustolle upotettava verkkomaksulomake, jolla voidaan tilata aikaan sidottuja palveluja. Tätä sovellusta kutsutaan tässä työssä verkkokaupaksi. Verkkokauppa-sovellus liittyy laajempaan ohjelmistoprojektiin, jonka Mindhive oy toteuttaa Mikkeli Siivotaan Oy:lle. Kyseisessä Web-sovelluksessa määritetään tilattava palvelu ja varataan sille sopiva ajankohta. Lopuksi annetaan tilaajan tiedot ja siirrytään maksamaan verkkomaksu tilaajan valitsemaan maksutavan palveluun. (Kuva 1.)



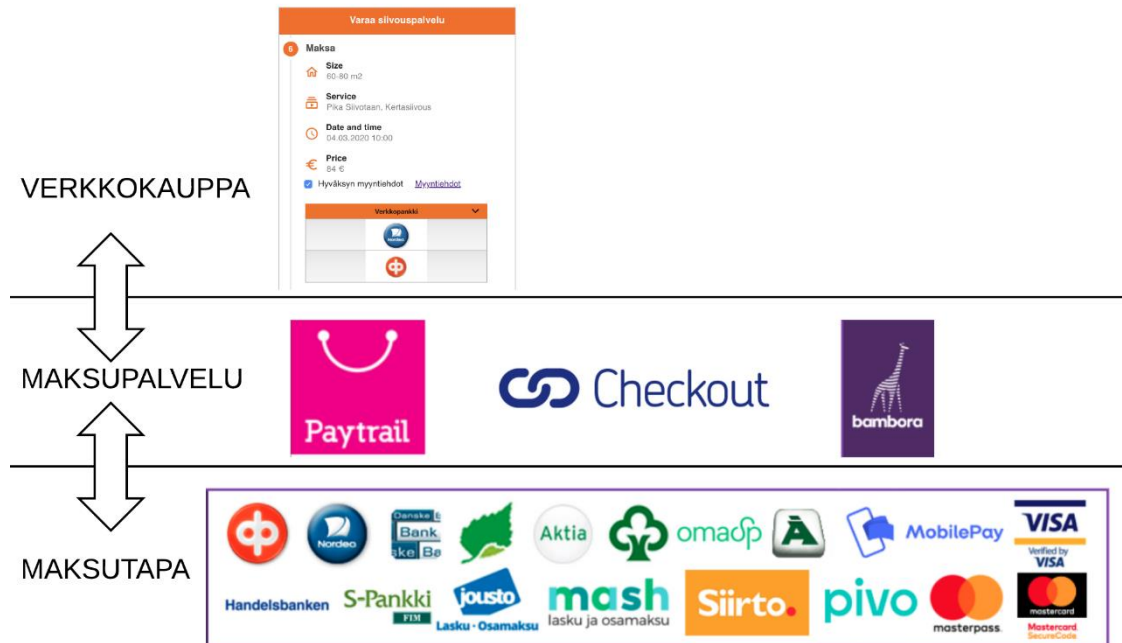
Kuva 1. Toteutettava verkkokauppa-sovellus

Onnistuneen maksutapahtuman jälkeen vahvistetaan tilauksen onnistuminen ja näytetään vielä kaikki tilauksen tiedot. Samalla asiakkaan tiedot ja tilaus tallennetaan palveluyrityksen tilauskalenteriin. Tässä opinnäytetyössä käsitellään tämän sovelluksen toteuttamiseen tarvittuja selvityksiä maksupalveluista ja verkkomaksun toteutuksesta.

2.2 Maksupalvelu ja maksutapa

Maksupalvelu on koonnut paketin eri maksutavoista ja tarjoaa sitä verkkokaupoille. Maksupalvelu toimii maksuliikenteen välittäjänä. Eri maksupalveluiden sivuilla käytetään maksupalvelusta eri nimityksiä: maksupalvelun tarjoaja ja maksupalvelun välittäjä. Lisäksi usein alkuliitteenä käytetään verkko sanaa, koska kyseessä on verkkomaksut, joita nämä verkkomaksupalvelut välittävät. Tässä työssä kutsutaan näitä toimijoita kuitenkin lyhyesti, nimityksellä maksupalvelu ja maksupalvelun toimija. Maksupalvelu on tehnyt sopimuksia eri

pankkien, luottokorttiyhtiöiden, luotollisten maksutapojen ja mobiilimaksamisen toimijoiden kanssa. Tässä työssä kutsutaan näitä toimijoita yhteisesti nimityksellä maksutapa ja maksutavan toimija. (Kuva 2.)



Kuva 2. Eri roolit verkkomaksussa

Eri maksupalvelut sisältävät suurelta osin samoja yleisimpiä maksutapoja ja ne eroavat toisistaan vain joiltakin osin. Ei ole syytä käyttää useaa maksupalvelua ainakaan samalle markkinointialueelle, koska asiakkuus maksaa.

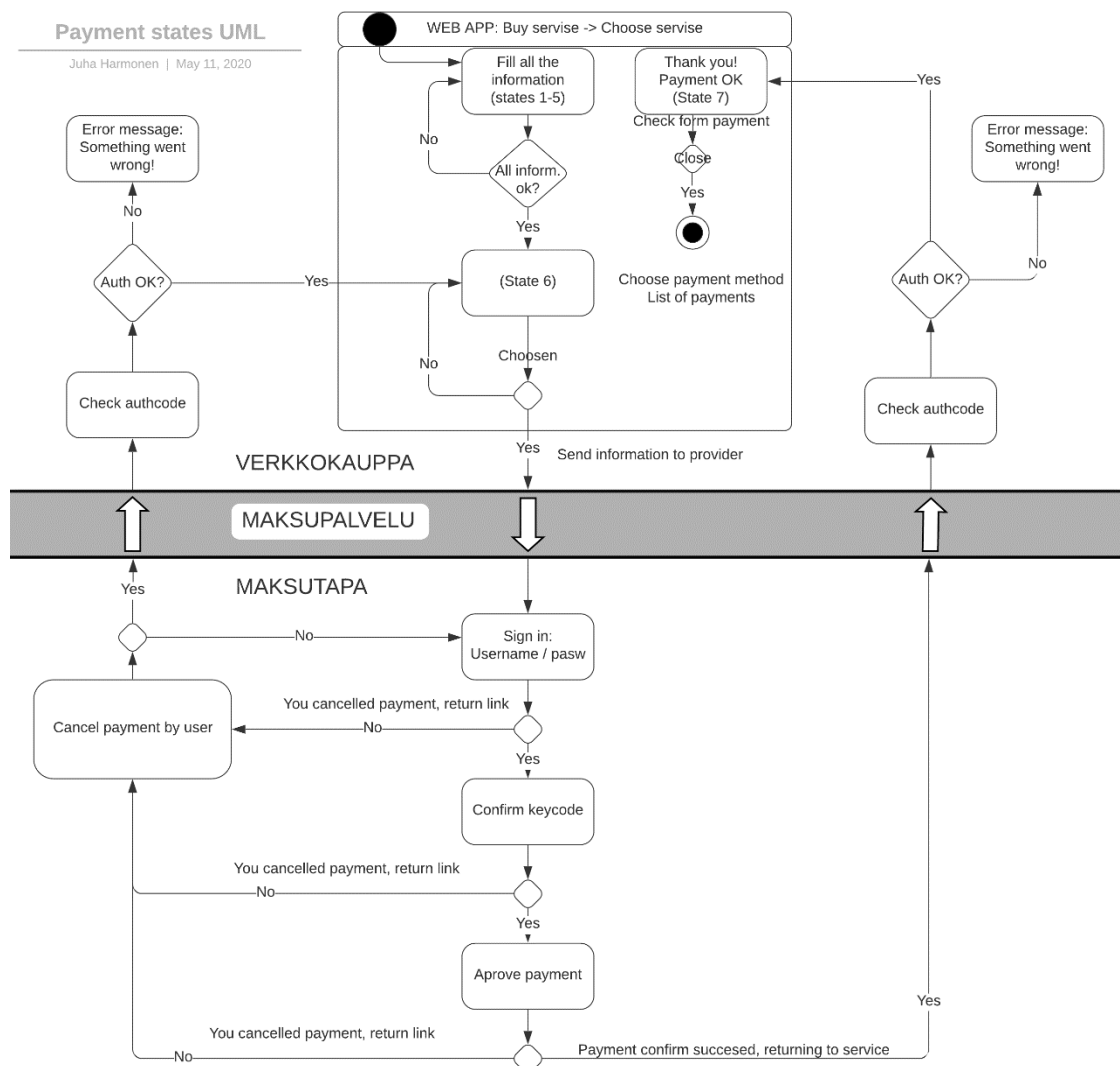
Maksupalvelu sekä välittää tiedot asiakkaan verkko-ostoksesta että ohjaa asiakkaan valittuun maksutapahtuman järjestäjän palveluun. Kun maksu on suoritettu, maksupalvelu ohjaa käyttäjän takaisin verkkokauppaan. Kaupan jälkeen maksupalvelu pitää maksutapahtumien tiedot tallessa, hoitaa API-raajapinnassa maksujen palautuksia ja vastaa maksutietojen kyselyihin.

Kun asiakkaan maksu hyväksytään maksutavan palvelussa, maksutavan toimija suorittaa rahansiirron tai varauksen asiakkaan tililtä. Maksutavan toimija välittää tiedon maksusta maksupalveluun. Maksupalvelu tallentaa maksutapahtuman tiedot omaan järjestelmäänsä ja vahvistaa maksun suorituksen verkkokaupalle. Kun verkkokauppa vastaanottaa tiedon maksusta, on hyvänä käytäntönä tehdä vielä varmistava kysely maksupalvelusta kyseisen maksun

tilasta. Tämän jälkeen maksu on hyväksytty, mutta rahaa ei ole siirretty muutoin kuin asiakkaan tililtä. Maksutavan toimija tilittää maksun maksupalveluun, oman käytäntönsä mukaan ja myöhemmin maksupalvelu tilittää sopimuksesta riippuen, 1 arkipäivän – 1 kk ajanjakson kaikki maksut verkkokaupalle. Maksupalvelu vähentää tilitettävästä summasta omat palkkionsa, tai tekee niistä oman laskun verkkokaupalle. (Tainio 2020.)

2.3 Verkkomaksun toteutuksen periaate

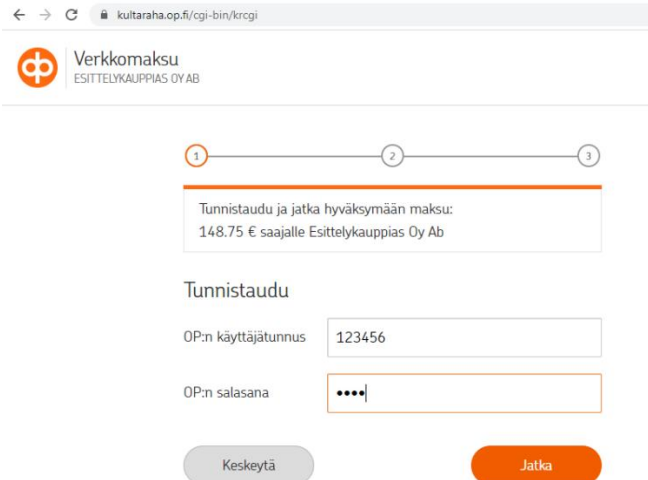
Aikaisemmin käytiin läpi eri toimijoiden roolit verkkomaksun toteutuksessa sekä kerrottiin toteutettavasta verkkokauppa-sovelluksesta. Tässä lähestytään asiaa toteutuksen kannalta, mitä viestejä välitetään eri vaiheissa. Kuvassa 3 nähdään verkkokauppa-sovelluksen periaate ja kuinka se kytkeytyy maksupalvelun avulla maksutavan palveluun.




Kuva 3. UML-kaavio verkkomaksusta verkkokauppa-sovelluksessa

Kuvan 3 kaaviossa kuvataan harmaalla palkilla, kuinka maksupalvelu yhdistää verkkokaupan ja maksutavan palvelun. Käyttäjälle maksupalvelu on lähes huomaamaton taustapalvelu. Kaavion keskellä olevalla rajatulla alueella kuvataan verkkokauppa-sovellusta. Sovelluksessa kerätään vaiheissa 1–5 (states) kaikki tilattavaan palveluun ja tilaajaan liittyvät tiedot. Vaiheessa 6 on maksusivu, jossa on näkyvissä kuvakkeet eri pankkien, pankkikorttien, mobiilimaksujen ja joustavien maksutapojen valinnalle. Käyttäjä valitsee haluamansa maksutavan, minkä jälkeen maksutapahtuman toteutus siirtyy valitun maksutavan toimijan nettipalveluun.

Kaaviossa olevan harmaan palkin alapuolella, on kuvattu Osuuspankin verkkopankkimaksun vaiheet ja mahdollisuudet keskeyttää maksutapahtuma. Seuraavaksi käydään läpi nämä maksun suorituksen vaiheet. Se alkaa siitä, kun käyttäjä ohjataan verkkokaupasta suoraan valitseman maksutavan palveluun. Käyttäjä kirjautuu tunnuksillaan (kuva 4).



← → ↻ kultaraha.op.fi/cgi-bin/krcgi

 Verkkomaksu
ESITTELYKAUPPIAS OY AB

1 2 3

Tunnistaudu ja jatka hyväksymään maksu:
148.75 € saajalle Esittelykauppias Oy Ab

Tunnistaudu

OP:n käyttäjätunnus

OP:n salasana

Kuva 4. Verkkomaksutapahtuman kirjautumisen 1. vaihe

Esimerkissä käytetyssä Osuuspankin verkkopalvelussa vahvistetaan tunnistautuminen avainlukulistan kysytylle koodille vastaavaa tunnusta (kuva 5).

Tunnistaudu ja jatka hyväksymään maksu:
148.75 € saajalle Esittelykauppias Oy Ab

Vahvista avainluvulla

Avainluku 0067

Jatka

Avainlukuja on jäljellä 100.

Keskeytä

Kuva 5. Verkkomaksutapahtuman kirjautumisen 2. vaihe

Yhtenä nykyisenä käytäntönä on, että pankki kertoo haluamansa avainlukulistan numeron tekstiviestillä käyttäjän puhelinnumeroon, käyttäjä lukee avainlukulistaa ja syöttää kysytyn tunnuksen (Aaltonen 2019). Tällöin henkilön vahvistus on kolmitasoinen: ensin pitää tietää käyttäjätunnus ja salasana, sitten hallussa on oltava tilinomistajan puhelin ja viimeiseksi on oltava painettu avainlukulista. Tunnistautumisen jälkeen käyttäjälle näytetään maksutapahtuman tiedot (kuva 6).

Hyväksy maksu

Maksun erittely Demo Yritys / 12345432

Viesti maksun saajalle DEMO YRITYS / 12345432

Saaja Esittelykauppias Oy Ab

Saajan tilinumero FI53 5000 0120 2018 67

Viitenumero 29727 28463

Eräpäivä 31.01.2020

Tililtä Testipäätili FI49 5000 9420 0287 30 110,49 €

Maksajan nimi TESTI ANNA

Määrä 148.75 €

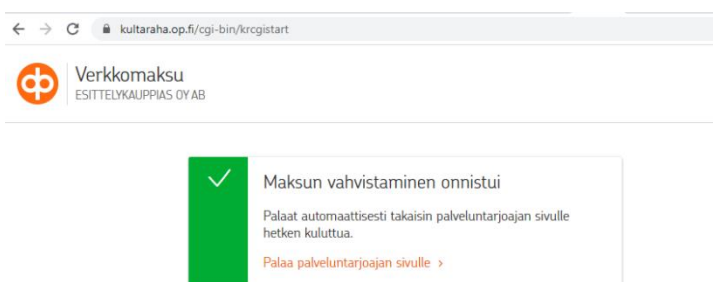
Maksun lisätiedot

Keskeytä Hyväksy

Kuva 6. Maksun hyväksyntä verkkomaksutapahtumassa

Käyttäjä voi valita tilinumeron, jolta summa veloitetaan. Pankki hoitaa valvonnan, että pankkitilin saldo on maksua suurempi. Tämä on käyttäjälle viimeinen

mahdollisuus keskeyttää maksutapahtuma. Huomaa, että kaikissa aikaisemmissakin ruuduissa on ollut keskeytä-painike, joiden painamisen jälkeen kysytään varmistus peruutukselle. Keskeytys johtaa siihen, että maksutavan palvelu peruuttaa maksutapahtuman ja välittää tiedon siitä maksupalvelulle. Maksupalvelu palauttaa käyttäjän verkkokauppaan ja kirjoittaa URL-riville maksun tiedot ja sähköisen allekirjoituksen tiedoista. Verkkokaupassa nämä tiedot luetaan ja tarkastetaan allekirjoituksen laskenta. Asiakas palautetaan maksutavan valintaan, josta voi yrittää maksun suoritusta uudelleen. Kun maksu onnistuu, pankki tekee maksun vähennyksen tililtä ja välittää maksun tiedot maksupalvelulle. Käyttäjälle näytetään hetken aikaa maksutavan palvelun kuittaus maksun onnistumisesta, ja siinä kerrotaan, että käyttäjä ohjataan automaattisesti palveluntarjoajan eli tässä tapauksessa verkkokaupan sivulle (kuva 7). Tämä on se hetki, jolloin maksu on onnistunut ja suoritettu. Vaikka käyttäjä sulkisi selaimen ikkunan, tämän muutaman sekunnin aikana, ennen kuin siirtyy takaisin verkkokauppaan, niin se ei maksua peru. Sulkemisesta aiheutuu se, että sitä ei voida heti merkitä verkkokaupassa maksetuksi, koska tieto maksun suorituksesta ei saavu verkkokauppaan uudelleenohjauksen keskeytyksen vuoksi. Tämä täytyy huomioida myöhemmin esimerkiksi backendissä ajettavalla vahvistamattomien maksujen tarkistuksella. Tästä lisää luvussa 5.3 Palvelun varausten käsittely.



Kuva 7. Maksun hyväksynnän vahvistus verkkomaksutapahtumassa

Kun uudelleenohjaus onnistuu, maksupalvelu palauttaa käyttäjän verkkokauppaan ja kirjoittaa URL-riville tiedot maksusta ja sähköisen allekirjoituksesta. Onnistuneessa maksussa tehdään samat tarkastukset ja tietojen ollessa todellisia tehdään vielä varmistava kysely maksusta backendillä, tallennetaan maksun tila ja näytetään tilausvahvistus käyttäjälle.

Maksu pitää suorittaa tietyn ajan kuluessa. Esimerkiksi Osuuspankin maksusuorituksen aikakatkaisu tulee tehtyjen testien perusteella jossain 12-15 minuutin välissä. Osuuspankki kysyy kolmeen eri kertaan tietoja, jokaisen vastausaikaa on maksimissaan 4-5 minuuttia. Eri maksutapojen palveluissa saat-
taa näissä olla eri käytäntöjä. Ajan ylittyminen käsitellään toimijoiden palveluissa samalla tavalla, kuin keskeytetty maksutapahtuma.

2.4 Verkkomaksun tietojen lähetys

Verkkomaksu alkaa siitä, kun verkkokauppa lähettää maksun tiedot maksupalveluun. Tiedot lähetetään maksupalveluun joko form-lomakkeena, tai API-rajapinnassa. Molemmat tavat on tarkasti kuvattu maksupalveluiden dokumentaatiossa ja tietojen on oltava määrittelyjen mukaisia. Verkkomaksun tietojen lähetys valmistelee tulevan maksutapahtuman. Esimerkiksi Paytrailin ja Checkoutin API-toteutuksissa nämä tiedot lähetetään verkkokaupasta heti kun käyttäjä on tullut sovelluksessa maksusivulle. Samalla käyttäjä tarkastaa rauhassa maksun tietoja ja valitsee maksutapaa. Verkkomaksu on tietyn aikaa voimassa ja maksun tunnus on myös kertakäyttöinen. Tunnisteen vanhenemisesta ilmoitetaan maksupalvelun sivulla ja käyttäjä voi palata linkistä takaisin verkkokauppaan. Jos tunnus vanhenee, täytyy muodostaa uusi tunnus lähettämällä maksun tiedot uudestaan. Verkkomaksujen tietojen lähetyksille ei ole rajoituksia ja käyttämättömänä ne vanhenevat. Kun käyttäjä painaa maksutavan painiketta, siirtyy painikkeen ja maksun tunnisteet maksupalveluun ja käyttäjä ohjataan maksutavan palveluun maksun tiedot mukana.

Verkkomaksun tietojen lähetys form-lomakkeena

HTML:ssä on määritelty form-elementti, jonka sisällä olevilla erilaisilla syöttökentillä (input) voidaan kysyä käyttäjältä tietoja. Tässä esimerkissä post tarkoittaa tietojen lähetystä ja action sisältää osoitteen, minne formin sisältämät tiedot lähetetään yhtenä pakettina. Kun kyseessä on maksutapahtuma, ei kuitenkaan haluta näyttää käyttäjälle kaikkia lomakkeen tietoja ja se onnistuu `type="hidden"`-määrittelyksellä. (Kuva 8.)

```
<h2>Paytrail maksulomake</h2>
<form action="https://payment.paytrail.com/e2" method="post">
  <input name="MERCHANT_ID" type="hidden" value="13466">
  <input name="URL_SUCCESS" type="hidden" value="http://www.exam.com/success">
  <input name="URL_CANCEL" type="hidden" value="http://www.exam.com/cancel">
  <input name="ORDER_NUMBER" type="hidden" value="123456">
```

Kuva 8. Verkkomaksun tietojen lähetys form-lomakkeena

Form-lomake on helpoin tapa toteuttaa verkkokaupan maksu ja käyttöliittymän teko on varsin suoraviivaista. Esimerkissä kaikki lomakkeen tiedot kerätään tietueeseen `paymentValues` ja tiedoista lasketaan Authcode-tieto (kuva 9).

```
import { sha256 } from 'js-sha256';


const strInformationShort: string = `${MerchantAuthHash}|
  ${paymentValues.MERCHANT_ID}|
  ${paymentValues.URL_SUCCESS}|
  ${paymentValues.URL_CANCEL}|
  ${paymentValues.ORDER_NUMBER}|
  ${paymentValues.PARAMS_IN}|
  ${paymentValues.PARAMS_OUT}|
  ${paymentValues.AMOUNT}`;

paymentValues.AUTHCODE = sha256(strInformationShort).toUpperCase();

console.log('AUTHCODE:', paymentValues.AUTHCODE);
```

Kuva 9. Esimerkki sähköisen allekirjoituksen laskennasta Reactilla

`MerchantAuthHash` arvo on salainen avain, jota käytetään ainoastaan sähköisen allekirjoituksen laskentaan, eikä sitä lähetetä formin tietojen mukana. Muut tiedot syötetään formin `value` -kenttiin seuraavan esimerkin mukaan: `<input name="AMOUNT" type="hidden" value={paymentValues.AMOUNT} />` Paytrailin maksupalvelussa form-lomake on yksinkertaisin keino lähettää verkkomaksu. Samalla käyttäjä uudelleenohjataan Paytrailin palveluun valitsemaan maksutavan (kuva 10).



Suomeksi | In English | På Svenska


TEST PAYMENT Use banks' TEST ACCOUNTS


PAYMENT DETAILS


Payment recipient/supplier:	Demo Yritys (Show information)
Order number:	123456
Total:	350,00 € (Show information)


SELECT PAYMENT METHOD




































Paytrail Oyj is shown as the payment recipient. The payment is settled to: Demo Yritys.

[Cancel payment](#)

Paytrail Plc is a payment institution which is supervised by the Finnish Financial Supervisory Authority.
 © 2007-2020 Paytrail Plc Business ID 2122839-7 www.paytrail.com



[Service description](#) | [Payment service provider's information](#) | [Safety information](#) | [Data privacy](#) | [Feedback](#)

Kuva 10. Maksutavan valinta Paytrailin palvelussa

Form-lomakkeen heikkoutena on tietoturva. Lomakkeella olevat piilotetutkin tiedot on helppo katsoa selaimen kehittäjien työkalujen avulla. Form-lomakkeen käyttö on vanheneva käytäntö ja edellä mainittujen syiden vuoksi se on jo poistunut esimerkiksi Bambora PayFormista.

Verkkomaksun lähetys API-rajapinnassa

API tarkoittaa ohjelmisto-rajapintaa eri ohjelmien välille. Tässä tapauksessa maksupalvelu on tehnyt omaan järjestelmäänsä sovelluksen, jolla on tarkoin määritelty avoin rajapinta. Verkkokauppa-sovellukseen tehdään määritysten mukainen yhteysmoduuli, joka välittää verkkomaksun tiedot maksupalvelun REST API -rajapintaan. REST tarkoittaa arkkitehtuurista kommunikointityyliä, erityisesti client-server-sovellusten tiedonsiirtoa internetin yli http(s)-protokollaa käyttäen (Kivisaari 2016). Kuvassa 11 on esimerkkikoodia maksun lähetyksestä Checkout verkko-API:ssa. Alkuperäinen Esa Nuurtamon Node.js koodi löytyy GitHubista (Nuurtamo 2019). Tässä esimerkissä koodia on muokattu funktioksi käyttäen TypeScriptiä.

```

import * as crypto from 'crypto';
import * as request from 'request-promise';
import * as uuidv1 from 'uuid/v1';

algorithm = 'sha256';
secret = secretStr;
merchantId = merchantIdStr;
baseUrl = 'https://api.checkout.fi';

const createPayment = async (payment: any) => {
  try {
    return await makeRequest(payment, 'POST', '/payments');
  } catch (err) {
    console.log('Error in create payment: ', err.message);
    throw err;
  }
};

const makeRequest = async (body: any, method: any, path: any) => {
  const timestamp = new Date();
  const checkoutHeaders: any = {
    'checkout-account': merchantId,
    'checkout-algorithm': 'sha256',
    'checkout-method': method,
    'checkout-nonce': uuidv1(),
    'checkout-timestamp': timestamp.toISOString(),
  };

  const url = baseUrl + path;
  const signature = calculateHmac(checkoutHeaders, body);
  const headers = { ...checkoutHeaders, signature, };

  const req = { method, url, headers, json: true, };

  let reqbody: any = req;
  if (body) reqbody = { ...req, body, };

  return await request(reqbody);
};

const calculateHmac = (params: any, body: any) => {
  const hmacPayload = Object.keys(params)
    .sort()
    .map(key => [key, params[key],].join(':'))
    .concat(body ? JSON.stringify(body) : '')
    .join('\n');

  return crypto
    .createHmac(algorithm, secret)
    .update(hmacPayload)
    .digest('hex');
};

```

Kuva 11. Maksun lähetys API:ssa. Muokattu Esa Nuurtamon Node.js GitHub-koodista.

Tietojen lähetys kutsutaan funktiolla `createPayment` ja siinä on parametrina maksun tiedot, dokumentin määritysten mukaan tehtynä. Se kutsuu edelleen `makeRequest`-funktioita, missä ensin muodostetaan headerit. Headerit ja maksun tiedot lähetetään edelleen `calculateHmac`-funktioille, joka laskee palautteena näiden tietojen allekirjoituksen. Allekirjoituksen tiedot lisätään headereihin ja muodostetaan paketti (`reqbody`) lähetettävistä tiedoista. Request-komento lähettää koko yhdistetyn paketin HTTP POST-komennolla `"https://api.checkout.fi /payments"`-osoitteeseen. API:n vastaus luetaan ja palautetaan loppujen lopuksi ensimmäisen `createPayment`-funktion kutsun teh-

neelle koodille. Checkoutin tapauksessa JSON-palautte sisältää maksutapahtuman tunnisteen, linkin checkoutin maksuvalintasivulle ja maksupainikkeiden tiedot parametreineen. Eri palveluiden palautteet ovat erilaisia, mutta yleensä ne sisältävät tilausnumeron, kertakäyttöisen nettilinkin ja maksun tunnisteen. Valittujen maksupalveluiden API-rajapinnat kuvataan tarkemmin maksupalveluiden esittelyssä luvussa 4.

2.5 Maksutapahtuman tietojen vastaanottaminen

Maksutapahtuman toteutumisen vahvistus ja siihen liittyvien tietojen vastaanottaminen toimii samalla tavalla, oli ostoksen tiedot välitetty maksupalveluun Form- tai API-rajapinnalla. Kun maksupalvelussa hyväksytään maksu, tai keskeytetään maksutapahtuma, uudelleenohjataan käyttäjä takaisin verkkokauppaan. Ohjaus tapahtuu kuvan 12 määrittämisen mukaan haluttuihin verkko-osoitteisiin. Määrittäminen sisältyy lähetettävään maksun tietoihin.

```
"redirectUrls": {  
  "success": "https://ecom.example.org/success",  
  "cancel": "https://ecom.example.org/cancel"  
},
```

Kuva 12. Esimerkissä määritellään käyttäjän uudelleenohjaus maksun jälkeen.

Verkkokauppa voi suunnitella haluamansa toiminnot kyseisissä tapahtumissa. Toteutettavassa verkkokauppa-sovelluksessa maksutapahtuman keskeytyminen johtaa maksuvalintasivulle uudelleen yritystä varten. Onnistunut maksutapahtuma taas kerrotaan verkkokaupassa, tallennetaan tarvittavat maksun tiedot tietokantaan ja esitetään kuitti ostotapahtumasta ruudulla. On kuitenkin ensin hyvä varmistaa, maksun suoritus vielä erillisellä kyselyllä Form- tai API-rajapinnasta.

Maksutapahtuman tiedot välitetään URL-osoitteen mukana. Tässä esimerkki:
`http://127.0.0.1:3000/payment/success?ORDER_NUMBER=12345432&TIMESTAMP=1580458504&PAID=15de5eb224&METHOD=2&RETURN_AUTHCODE=182E176A5903101F4D7B5767ED733916`

Kysymysmerkin jälkeiset tiedot voidaan ohjelmallisesti hakea parametreinä. Nämä parametrit ovat esimerkiksi Paytrailin form-lomakeversiossa määritetty `PARAMS_OUT` tiedossa. Eli sen kentän tietoihin laitetaan halutut paluutiedot.

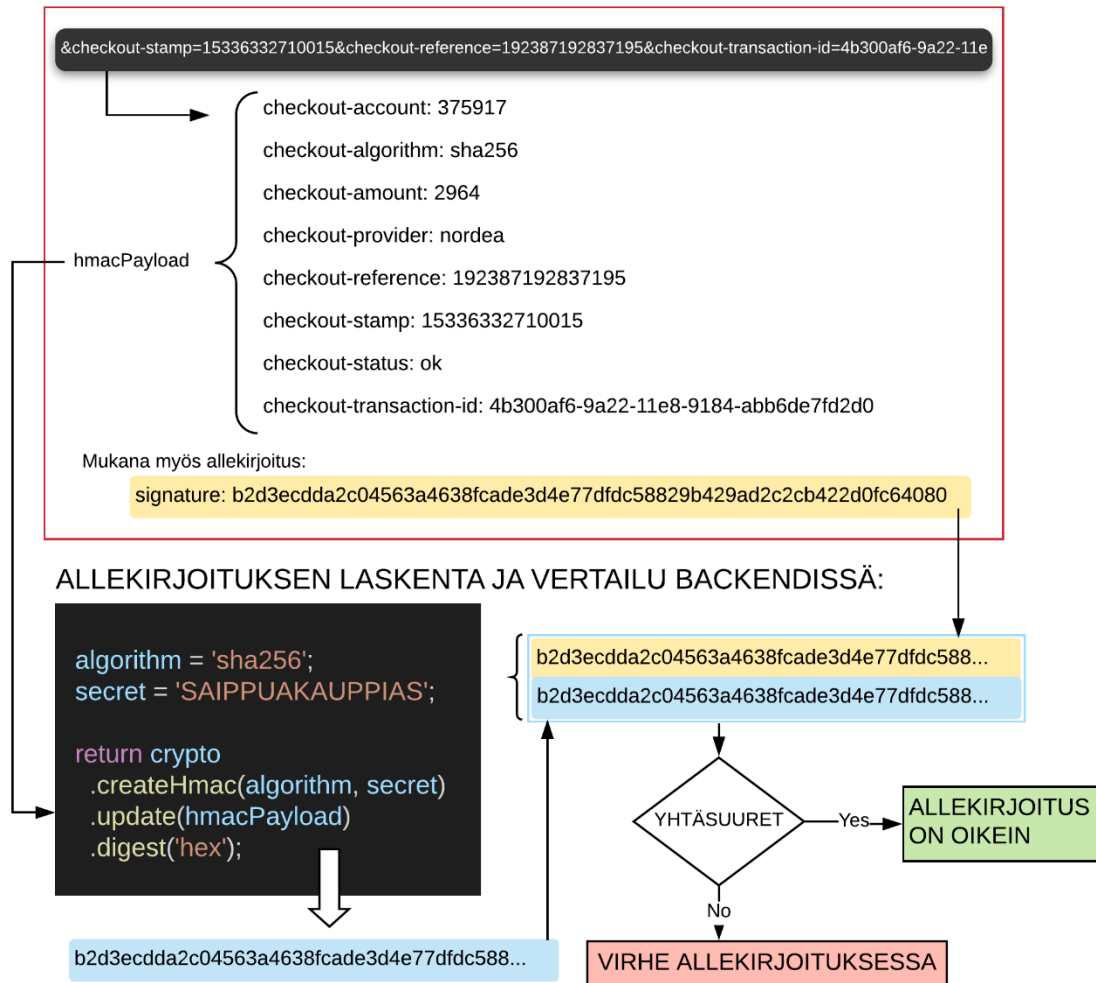
Pakollisia tietoja ovat PAYMENT_ID, TIMESTAMP, STATUS ja RETURN_AUTHCODE. Kaikki mahdolliset Paytrailin tiedot löytyvät kuvassa 13.

Field name	Description	Notes
ORDER_NUMBER	This is the same order number that was sent to the Payment Gateway.	
PAYMENT_ID	Unique id for payment.	Value is always included.
AMOUNT	Payment total	AMOUNT equals the value provided in payment creation.
CURRENCY	Currency.	
PAYMENT_METHOD	Payment method which was selected at payment method page (or preselected with value PAYMENT_METHODS).	Value is empty if no payment method has been selected.
TIMESTAMP	Timestamp generated by the Payment Gateway that is used to calculate the return authentication hash. Timestamp is in UNIX format, i.e. seconds from 1/1/1970.	Value is always included.
STATUS	It is used to verify the validity of a successful payment.	Possible values PAID and CANCELLED.
SETTLEMENT_REFERENCE_NUMBER	This is the same reference number that was sent to the Payment Gateway or the reference number generated by Paytrail.	
RETURN_AUTHCODE	If the value matches the calculated one, the payment has been completed and the information has not been modified after sending.	Value is always included.

Kuva 13. Paytrailin dokumentaation määrittäminen URL-parametreista

Viimeinen tieto on RETURN_AUTHCODE. Tätä tietoa meidän täytyy käyttää, jotta varmistetaan, että kuitin kaikki tiedot ovat todellisia ja ne ovat välittyneet muuttumatta. Kyseessä on viestin sähköisestä allekirjoituksesta (Microsoft 2010). Maksupalvelu laskee allekirjoituksen kaikista lähetetyistä URL-parametreista. Allekirjoituksen laskennassa käytetään kyseiselle verkkokaupalle annettua omaa salaista avainta ja avain on siis vain maksupalvelun ja verkkokaupan tiedossa. Verkkokauppa tekee saman laskennan viestin tiedoilla ja salaisella avaimella, ja vertaa vastaanotettua allekirjoitusta itse laskettuun. Kun allekirjoitukset ovat samat, voi tietoihin luottaa. Kaaviossa salainen avain on muuttuja nimeltään secret ja sen arvo on "SAIPPUAKAUPPIAS". Salaisen avaimen vuoksi laskenta suoritetaan backendissä (luku 2.6). Tässä Checkoutin esimerkissä on käytetty todellisia arvoja ja siinä on kaikki tällä hetkellä käytössä olevat checkout-alkuiset parametrit aakkosjärjestykseen järjestettynä, kuten ne laskennassa täytyy olla. Checkoutin dokumentaatiossa kerrotaan, että parametreja ei saa kovakoodata, sillä niitä saattaa tulla lisää. (Kuva 14.)

MAKSUPALVELU LÄHETTÄÄ URL-PARAMETRIT VERKKOKAUPPAAN



Kuva 14. Digitaalisen allekirjoituksen varmistaminen Checkoutissa

Kuvan 11 esimerkissä näkyy, kuinka samaa allekirjoituksen laskentaa käytetään maksun tietojen lähetyksen yhteydessä. Siinä esimerkissä on laskennan suorituksen koodit esitetty kokonaisuudessaan. Huomioitavaa siinä on se, kuinka tiedot on järjestetty sort-toiminnolla ja kuinka kaikki tiedot yhdistetään toisiinsa.

Tehdään vielä esimerkki laskennasta Paytraililla. Kuvassa 15 on Reactilla tehty esimerkki. Sen `readUrl`-funktiossa ensin luetaan URL-parametrit muuttujiin. Ne yhdistetään dokumentaation mukaiseen järjestykseen ja lisätään väleihin `'|'`-merkki. Parametrien lisäksi lisätään `MerchantAuthHash`-muuttujan arvo, minkä arvo on Paytrailin kyseiselle verkkokaupalle antama salainen avain. Tästä `checkString`:stä lasketaan SHA256-salauslaskenta ja verrataan tulosta `RETURN_AUTHCODE`-parametriin. Kun ne ovat yhtä suuret, on allekirjoitus oikein.

```

const [codeOK, setCodeOK,] = useState<boolean>(false);
const [receipt, setReceipt,] = useState<string>('');

const countSHA256 = (str: string) => {
  return sha256(str).toUpperCase();
};

const readUrl = () => {
  const params = new URLSearchParams(location.search);
  if (params.has('PAYMENT_ID')) paymentId = params.get('PAYMENT_ID');
  if (params.has('TIMESTAMP')) timestamp = params.get('TIMESTAMP');
  if (params.has('STATUS')) status = params.get('STATUS');
  if (params.has('ORDER_NUMBER')) orderNumber = params.get('ORDER_NUMBER');
  if (params.has('AMOUNT')) amount = params.get('AMOUNT');
  if (params.has('RETURN_AUTHCODE')) returnAuthcode = params.get('RETURN_AUTHCODE');

  const checkString: string = `${orderNumber}|${paymentId}|${amount}|${timestamp}|${status}|${MerchantAuthHash}`;
  if (countSHA256(checkString) == returnAuthcode)
    console.log('Koodi on oikein!');
    setCodeOK(true);
    setReceipt(`${orderNumber} ${paymentId} ${timestamp} ${status} ${amount}`);
};

```

Kuva 15. React esimerkki Paytrailin RETURN_AUTHCODEN tarkastuslaskennasta

Jos RETURN_AUTHCODE on sama kuin salauksen tarkastuslaskennassa saadaan, voidaan käyttäjää kiittää ostoksesta ja näyttää kuitin tiedot. Jos tarkastuslaskenta tuottaa eri tuloksen, täytyy näyttää virheilmoitus käyttäjälle ja välittää tieto verkkokaupan ylläpitoon. Maksutapahtuma täytyy tarkistaa, ennen kuin se voidaan merkitä maksetuksi verkkokaupan tietokantaan. Paytrailin dokumentaation esimerkissä (Paytrail Example 4.12) tämä tulkitaan seuraavasti: "Maksun kuitti ei ollut kelvollinen, mahdollinen maksupetoksen yritys" tai, kuten heidän ladattavassa PHP-esimerkissä: "Virheellinen ilmoitus, mahdollisesti joku yrittää hakkeroida sen. Älä tee mitään tai luo hälytys." Tämä kannattaa ottaa vakavasti, tallentaa virhetapaukset ja seurata niitä. Järjestelmän väärinkäyttö ei ole kuitenkaan helppoa. Kaikki tiedot, jotka tarvitaan RETURN_AUTHCODE:n laskentaan, eivät ole URL-osoitteen mukana.

2.6 Maksutapahtuman tietoturvallinen käsittely

Maksutapahtuman ja sen tietojen varmistaminen ovat tärkeimpiä asioita verkkokaupalle. Yhtään maksumerkintää ei saa syntyä ilman vastikkeen vaihtoa. Verkkokaupan myyntimäärät tulee olla yhteneviä tilitettäviin maksuihin. Näiden seurannalla voidaan jälkikäteen varmistaa, että petoksia ei ole tapahtunut. On kuitenkin viisasta varmistaa, että ohjelmisto toteutetaan turvallisesti. Verkkokaupat ovat verkkosovelluksia, ja ne toteutetaan selainpohjaisella käyttöliittymällä. Selaimen koodeja voidaan tutkia, joten niissä ei saa olla salaista tietoa saatavilla. Selain on haavoittuva ja sitä kautta voidaan tehdä huijaus yrityksiä.

Siksi verkkokauppa pitää toteuttaa siten, että selaimen (frontend) koodit eivät sisällä arkaluontoisia toimintoja, vaan se välittää tiedot ja kyselyt serverille (backend), jossa tehdään laskennat, varmistukset ja tarvittavien tietojen tallennukset.

Verkkokaupan tarkistettavat tietoturva-asiat

Maksupalvelusta saadaan asiakkuuden tunniste ja salainen avain. Näitä tarvitaan tietojen lähetyksessä ja vastaanotossa, sähköisen allekirjoituksen laskentaan. Nämä tiedot tulee käsitellä vain serverillä ja lisäksi ne on pidettävä tietoturvallisesti hallussa. Tunnuksia ei laiteta ohjelmakoodiin suoraan, vaan ne haetaan tiedostosta tai tietokannasta. Yksi mahdollisuus arkaluonteisen tiedon varastointiin on käyttää dotenv-kirjastoa, joka mahdollistaa tietojen säilömiseen erilliseen .env-tiedostoon (environment variables eli ympäristömuuttujat). Jos yrityksen koodit ovat GitHubissa tai muussa vastaavissa palvelussa ja tunnukset ovat erillisessä tiedostossa, pitää lisätä tämä salaista tietoa sisältävä tiedosto .gitignore-tiedostoon. Varmistukseksi tallennettavat tunnukset kannattaa salata eri tekniikoin. Tällöin salattujen tunnusten paljastuminen, ilman salauksen purkuavainta, ei ole kovinkaan suuri riski. Tässä työssä toteutettavassa sovelluksessa on käytössä Googlen pilvipalvelut. Siinä tunnisteet voi tallentaa tietoturvallisesti käyttäen Cloud Key Management Serviceä. Se on pilvipalvelu, jossa voi luoda, tuoda ja hallita salausavaimia. Lisäksi yksittäisten avainten käyttöoikeuksia voidaan hallita ja valvoa.

Maksutapahtuman jälkeen selaimen URL-kenttään tulee maksutapahtuman vahvistus. Tämä ei ole riittävä tieto maksun hyväksynnän tallentamiseen, vaan sen jälkeen lähetetään serverillä kysely maksutapahtumasta maksupalveluun. Jos vastaukseksi saadaan tiedot hyväksytystä maksusta, nämä tiedot tallennetaan serveriltä suoraan tietokantaan.

Jos sovelluksessa liikkumiseen käytetään URL-kentän tietoja, on huomioitava, että mitään tärkeää tietoa ei tallenneta sivuston URL-osoitteeseen. Tämä siksi, että kaikki kohdennettu tieto voi auttaa tietomurtojen tekemisessä. URL-kentän tiedot voi myös tyhjentää aina kun niitä ei tarvita enää. Tällöin tiedot vain vilahtavat ja niiden tutkiminen vaikeutuu. Tätä parempi ratkaisu on kuitenkin salata tiedot, jolloin tietojen purkamiseen tarvitaan salainen avain.

3 MAKSUPALVELUIDEN VERTAILU MINDHIVE OY:LLE

Tässä luvussa verrataan eri maksupalveluja. Tehdään huomioita niiden eroista ja painotetaan mitkä ovat haluttuja ominaisuuksia Mindhiven kannalta. Ehdokkaita karsitaan pois, jotta voidaan keskittyä myöhemmissä testeissä vain sellaisiin, jotka voitaisiin ottaa sovelluksessa käyttöön. Karsintaan riittää yksikin kriteeri, jos se katsotaan tässä tapauksessa merkitseväksi perusteeksi.

Kaikki tässä esiteltävät maksupalvelut ovat laajasti käytössä ja ne soveltuvat verkkomaksujen suoritukseen hyvin. Eroja on mm. maksuvaihtoehdoissa, sopimusten kestossa ja hinnoittelussa. Eroja on myös mille verkkokauppa-alueille niillä on liityntä ja onko niillä avoin API-rajapinta omien verkkokauppasovellusten toteuttamiseen. On huomioitavaa, että kaikki tiedot perustuvat kirjoitushetken tilanteeseen. Tällä rintamalla tapahtuu koko ajan, uusia toimijoita tulee ja vanhat muuttavat muotoaan. Maksutapojen trendit muuttuvat ja ne vaikuttavat eri toimijoihin eri aikataululla. Yleisesti sanoen toimijoiden on tarjottava kattava määrä eri maksutapoja kilpailukykyiseen hintaan, muutoin asiakkaat siirtyvät muualle. Maksupalvelun vaihtaminen ei ole teknisesti suuri projekti, koska maksupalveluiden tekniset toteutukset ovat hyvin lähellä toisiaan. Kun verkkokauppa on tehty yhden toimijan API-rajapintaan, pystyy samasta ohjelmistoprojektista tekemään uuden version, joka keskustelee uuden rajapinnan kanssa.

3.1 Maksuvaihtoehtojen vertailu

Mindhive Oy:n kriteereitä maksuvaihtoehdoille ovat monipuoliset maksutavat, kansainvälisyys, laskulla maksamisen mahdollisuus ja kuukausittain toistuvat maksut. Monipuolisissa maksutavoissa ja kansainvälisyydessä viitataan sekä Suomen että lähialueiden suosituimpiin maksutapoihin. Laskulla maksaminen sopii hyvin yrityksille ja automaattisesti toistuvat maksut ovat käteviä sekä myyjälle että ostajalle.

Tärkeintä maksupalveluissa ovat monipuoliset maksutavat, jotta asiakkaat löytävät haluamansa tutun maksutavan. Sen vuoksi maksutavat aloittavat toimijoiden karsimisen. Seuraavassa taulukossa on vertailu 8 palveluntarjoajan eri maksutavat (Huttunen 2019).

	Korttimaksu	Verkkopankki	Lasku tai osamaksu	Mobiilimaksu	PayPal tai vastaava	Toistuvat maksut
	✓	✓	✓	✓	✗	✓
	✓	✓	✓	✓	✓	✗
	✓	✓	✓	✓	✗	✗
	✗	✗	✓	✗	✗	✗
	✓	✓	✓	✗	✗	✗
	✓	✗	✗	✗	✓	✓
	✓	✓	✓	✓	✓	✗
	✓	✗	✗	✓	✓	✓

Kuva 16. Verkkokauppojen maksutavat taulukko (Huttunen 2019)

Kuvasta 16 nähdään, että Mash ja PayPal ovat varsin yksipuolisia palveluja ja kummastakin puuttuu verkkopankkimaksaminen, mikä tiputtaa ne vertailusta pois. Tämä siksi, että Paytrailin tutkimuksen mukaan Suomessa verkkopankkimaksaminen on tärkein maksutapa (Verkkokauppa Suomessa 2019, 20-21.). Tässä on kuitenkin tapahtunut muutosta ja pankki- tai luottokortti maksut ovat PosNordin tutkimuksen mukaan kasvattanut suosiotaan ja on kirjinyt jo verkkopankkimaksamisen ohi (Verkkokauppa Pohjoismaissa 2019, 34). Nämä ovat joka tapauksessa kaksi tärkeintä maksutapaa Suomessa (taulukko 1). Merkittävämpi maksutapa tässä tapauksessa on verkkopankkimaksaminen, koska se on näissä palveluissa harvinaisempi. Näissä vuosittain tehtävissä raporteissa PostNord kysyi millä asiakkaat maksavat verkko-ostoksensa, kun taas Paytrail kysyi mieluisinta maksutapaa (Huttunen 2019).

Taulukko 1. Mitä maksutapoja suomalaiset suosivat? (Huttunen 2019)

	PostNord	Paytrail
Verkkopankilla	27 %	39 %
Pankki- tai luottokortilla	30 %	24 %
Laskulla	25 %	16 %
PayPalilla, Paysonilla tai vastaavalla	12 %	13 %
Käteismaksulla	2 %	0 %
Osamaksu	0 %	4 %

Klarnassa on käytössään omat verkko- ja korttimaksutavat, mutta sillä ei ole saatavilla Mobiilimaksua, Paypalia eikä toistuvia maksuja. Näiden puutteiden vuoksi tämä toimija tiputetaan tästä joukosta pois. Eniten maksutapoja on

Bambora PayFormissa, BlueCommercessa, Checkoutissa, Paytrailissa ja Stripessa. Stripe pudotetaan kuitenkin myös pois verkkopankkimaksutavan puuteen vuoksi. BlueCommerce toimittaa vain verkkokauppa-moduuleita, joten sillä ei ole avointa API-rajapintaa, jota tarvitaan tässä toteutuksessa.

Apple Pay ja Google Pay ovat uusia maksupalveluita. Molemmat tarjoavat eri pankkikorttien ja luottokorttien maksutapaa, joissa kortin tiedot tallennetaan heidän palveluunsa. Näin ne ovat käytettävissä eri verkkokaupoissa suoraan. Monet verkkokaupat tarjoavat samaa kortin tietojen tallennuspalvelua, mutta tällöin tallennus on verkkokaupakohtainen. Korttitietojen tallennusta pidetään sekä turvallisempaan että helpompaan kuin, että korttitiedot syötetään aina maksun yhteydessä. Apple Payn ja Google Payn kasvulle on odotusarvoa, koska nämä yritykset hallitsevat varsinkin mobiililaitteita ja niiden käyttäjiä sovellusympäristöillään. Käyttäjät on veloitettu kirjautumaan näiden toimijoiden palveluun, jotta mobiililaitteita voidaan käyttää. Tästä on vain pieni askel siihen, että kirjautuu myös heidän maksutapoihinsa. Tällä hetkellä tämän maksutavan tarjoamista ei ollut havaintoa kyseisissä maksupalveluissa, mutta Checkoutille ne ovat tulossa. Tätä rajoittaa tällä hetkellä se, että eri pankkien täytyy tehdä suoraan sopimuksia Applen ja Googlen kanssa. Pelkät luottoyri-tysten, kuten Visa ja Mastercard, tekemät sopimukset eivät riitä.

Maksuvaihtoehtojen käsittelyn jälkeen, jäljelle jäi Bambora PayForm, Checkout ja Paytrail. Niiden maksutavat ovat monipuoliset ja ne soveltuisivat hyvin Mindhiven käyttöön. Seuraavissa luvuissa vertaillaan vain näitä toimijoita.

3.2 Hinnoittelujen vertailu

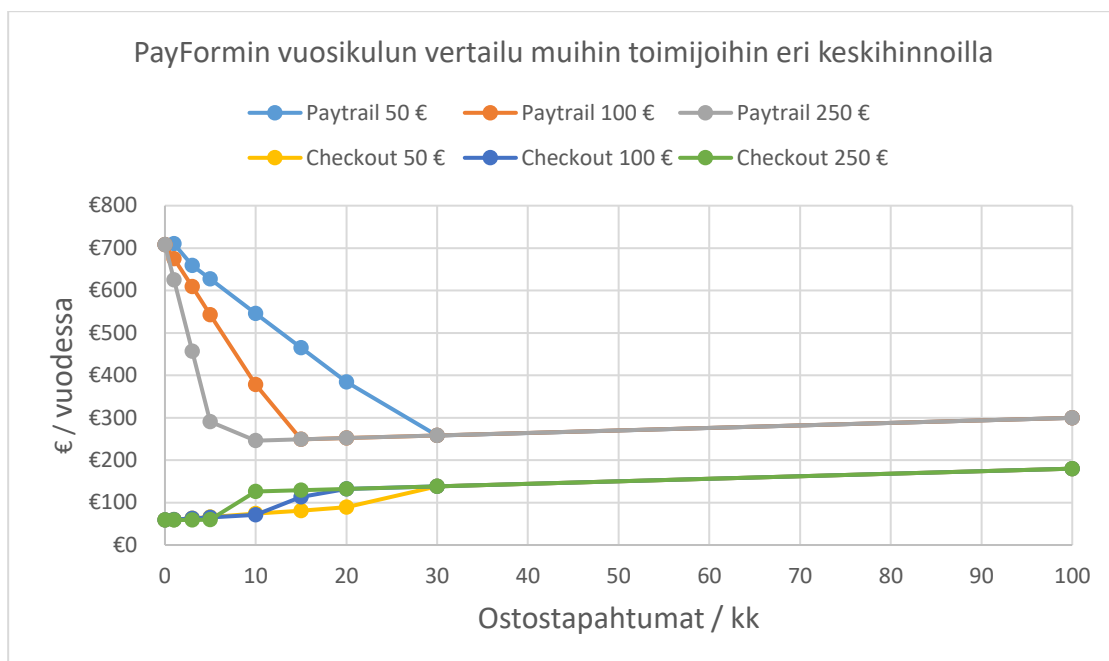
Mindhive Oy:n kriteerit hinnoittelun osalta ovat kilpailukykyinen hinta ja se, että hinnoittelu soveltuu myös pieneen verkkokauppaan. Tämän luvun esitetyt hinnat perustuvat nettisivuilla kerrottuihin hintoihin ilman lisäpalveluita (päivitetty 1.2.2020). Joidenkin toimijoiden nettisivuilla opastetaan ottamaan myyntiin yhteyttä ja kysymään tarjousta. Yksi tapa tutkia hinnoittelua on se, että onko palvelussa kiinteä kuukausihinta (Huttunen 2019). Jos verkkokaupan käyttö on ollut pientä, tai sen odotetaan olevan pientä, kannattaa valita kuukausimaksuton vaihtoehto.

	Kuukausimaksuton vaihtoehto
	✓
	✓
	✗
	✓
	✓
	✓
	✗
	✓

Kuva 17. Kuukausimaksuttomat maksupalvelut (Huttunen 2019)

Checkoutissa ja Paytrailissa on aina kuukausimaksu (kuva 17). Checkoutissa on valittavissa hinnoittelu, jossa on pieni kuukausimaksu. Jos verkkokauppa on pienimuotoista tai epävarmaa, Paytrail ei ole paras aloituskumppani suuren kuukausimaksun vuoksi. Vaikka muissa palveluissa on valittavissa kuukausimaksuton palvelu, on niihin osaan valittavissa myös palvelusopimuksia, joissa on kuukausimaksu. Kun verkkokauppa tapahtumien määrä kasvaa kannattaa siirtyä näihin kalliimpiin kuukausimaksullisiin sopimuksiin. Säästöä tulee sekä tapahtumahinnoissa että provisioissa. Jos aloittaa kuukausimaksuttomalla sopimuksella, täytyy vaihtaa sopimuksen tyyppiä, kun verkkokauppa kasvaa. Muutoin halpa sopimus tulee kalliiksi. Maksuvertailusta ei voi tehdä aivan suoria johtopäätöksiä, koska muuttuvia tekijöitä on paljon. Verkkokaupan asiakas valitsee maksutavan, joten heidän toimintansa ei ole ennakoitavissa muutoin, kuin aiheesta tehtyjen tutkimusten perusteella.

Paytrailin hinnoittelu perustuu kiinteään kk-hintaan ja yksittäisten maksutapahtumien tapahtumahintaan ja provisioon luottokorttimaksuissa. Checkout tarjoaa kolme eri asiakkuuden vaihtoehtoa, eri kokoisten verkkokauppojen tarpeisiin. Täten Checkout skaalautuu helpommin pieneen verkkokauppaan. Kun ostosten määrä lisääntyy, voi sopimusta vaihtaa. Checkout hinnoittelu eroaa Paytrailista kk-hinnan lisäksi hieman laajemman provisiohinnoittelun osalta. Provisiota otetaan myös pankkimaksuissa ja mobiilimaksamisessa. Näillä on merkitystä, jos ostosten keskihinnat ovat suurempia.



Kuva 18. PayForm on halvin

PayForm on hivenen muita toimijoita halvempi (kuva 18). Kaaviossa muiden vuosikuluista on vähennetty PayFormin hinta, jolloin kaavio kertoo paljonko muut ovat vuositasolla kalliimpia. Laskennassa on optimoitu eri asiakkuus vaihtoehdot, kussakin tilanteessa, on valittu kaikkein halvin sopimus. Esimerkiksi, jos keskimääräinen ostos on 50 €, niin tapahtumien määrän saavuttaessa 30 kpl / kk (1500 € / kk), kannattaa sopimus vaihtaa Checkoutissa Startista Laajaan ja PayFormissa Expressistä Poweriin. Jos keskihinta on 100 €, 16 ostosta / kk tai enemmän kannattaa hoitaa Checkoutin Laajalla ja PayFormin Power sopimuksella. 250 € keskihintaisille ostoksilla raja on jo niin pieni kuin 7 ostosta / kk. Helppointa on kuitenkin valvoa liikevaihtoa. Checkoutilla liikevaihdon raja on n. 1600 € / kk, jolloin kannattaa siirtyä Startista Laajaan sopimukseen. Bambora PayFormilla raja on n. 1500 € / kk, jolloin kannattaa siirtyä Expressistä Poweriin. Checkout Startti ja PayFormin Express paketit toimivat vain pienille ostoksille ja pienille ostostapahtumien määrille. Kun verrataan Paytrailia Checkoutin Laajaan pakettiin, on eroja lähinnä vain kuukausimaksuissa ja mobiilimaksuissa. Bambora PayForm oli halvin sekä kuukausimaksussa että jokaisessa maksutavassa. Erot ovat kuitenkin pieniä. Jos esimerkiksi ostostapahtumia on 100 kpl/kk ja keskiostos 100 €, niin erot vuosikuluissa on 180 – 300 €/vuodessa. Se on vähän suhteutettuna 120 t€ liikevaihtoon. Tällöin PayForm on 0,25 % ja Checkout 0,15 % kalliimpi kuin Bamboran PayForm. Tämä kertoo, että kilpailu toimii alalla.

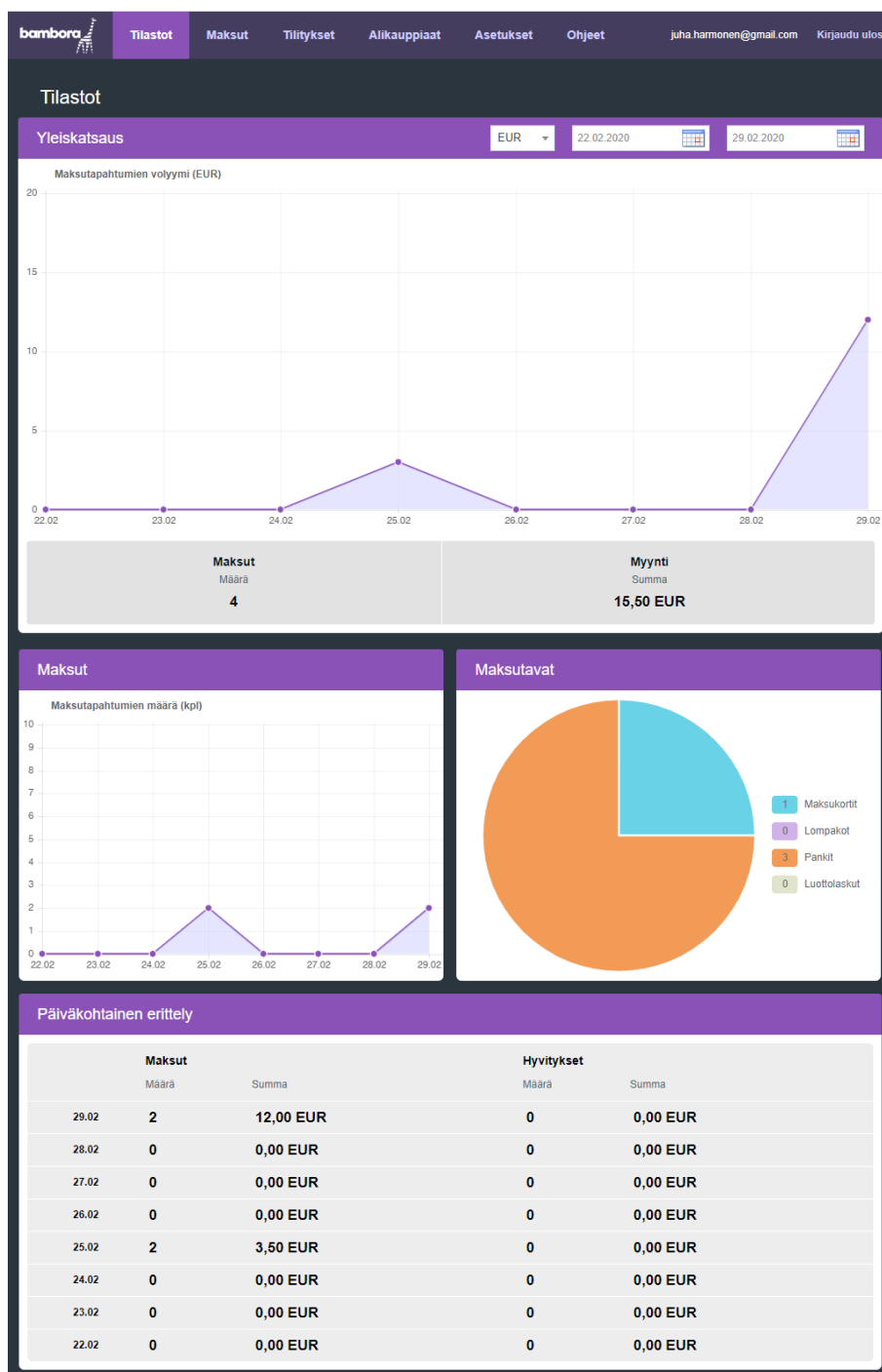
Hinnoittelun vertailu ei tuota tulosta maksupalvelun valintaan, kuin vain siinä tapauksessa, että verkkokauppa on hyvin pientä. Tällöin Paytrail voi tuntua kalliilta. Hintaeroon vaikuttaa myös se, mitä lisäpalveluja tarvitsee eri toimijoille. Jotta maksutapahtuma sujuu mahdollisimman suoraviivaisesti, halutaan että maksuvalinnan painikkeet ovat verkkokaupan käyttöliittymässä. Tämä toiminto on näissä kaikissa maksupalveluissa, mutta Paytrail on ainut, jossa maksuvalintasivun ohitus on maksullisena lisäpalveluna.

3.3 Maksupalveluiden maksutapahtumien käsittelyn sovellukset

Verkkokaupan pitää hoitaa palautuksista aiheutuvat hyvitykset. Verkkokauppa haluaa myös valvoa maksutapahtumia ja tulostaa raportteja. Näihin toimintoihin löytyy API-rajapinta, joka toimii maksusuoritusten tunnisteella tai asiakasnumerolla. Rajapinnan käyttö voi olla mielekästä, jos halutaan siirtää automaattisesti maksutapahtumien tietoja toiseen järjestelmään. Maksupalvelut antavat käyttöön myös ilmaiset verkkosovellukset, joilla nämä halutut toiminnot kannattaa suorittaa. Kaikkien toimijoiden sovellukset eivät toimi testitunnuksilla. Tässä esitetään ominaisuuksia, osin valmistajan sivuilta löytyneiden dokumenttien perusteella. Eri toimijoiden sovellukset on nimetty eri nimillä, mutta kaikissa on kauppias-alkuliite. Tässä työssä viitataan kaikkiin näihin sovelluksiin Paytrailin käyttämällä suomenkielisellä sanalla kauppiaspaneeli.

Kauppiasportaali (Bambora PayForm)

Bambora on ainut, jonka sovellus toimii testitunnuksilla. Sovellus näyttää etusivulla hienoa grafiikkaa tilastoista ja uudet tapahtumat näkyvät siinä reaaliaikaisesti. Tilastot voi hakea haluamaltaan ajanjaksolta. (Kuva 19.)



Kuva 19. Bambora PayFormin kauppiasportaalin etusivu

Yksittäiset maksut näkevät maksut sivulta (kuva 20). Näytä hakuvalinnat-painike tuo monipuoliset hakukentät, joilla saa rajattua haluamansa tulokset. Tulokset voi viedä CSV-tiedoston avulla Exceliin.

Luotu	Tila	Alikauppias	Tilaustunnus	Summa
29.02.2020 16:21	Hyvitys hyväksyty	Mindhive Oy TEST	test-order-2-1582960227998	-2,00 EUR
29.02.2020 11:45	Varmennus epäonnistui	Mindhive Oy TEST	test-order-3-1582965938155	6,00 EUR
29.02.2020 10:10	Veloitettu	Mindhive Oy TEST	test-order-2-1582960227998	6,00 EUR
29.02.2020 10:03	Veloitettu	Mindhive Oy TEST	test-order-1-1582959784056	6,00 EUR
25.02.2020 12:42	Veloitettu	Mindhive Oy TEST	test-order-1-1582623728003	2,50 EUR
25.02.2020 12:22	Varmennus epäonnistui	Mindhive Oy TEST	test-order-2-1582622544736	1,00 EUR
25.02.2020 12:20	Veloitettu	Mindhive Oy TEST	test-order-1-1582622416792	1,00 EUR

Kuva 20. Bambora PayFormin kauppiasportaalin maksut välilehti

Maksuriviä klikkaamalla pääsee maksun yksityiskohtiin, jossa maksulle voi tehdä mm. hyvityksen. Maksun toteutumisesta löytyy tarkat tiedot aikaleimoineen. (Kuva 21.)

Tuotenumero	Nimike	Veroton hinta (kpl)	ALV-prosentti	Hinta (kpl)	Lukumäärä	Kokonaishinta	Tuotteen tyyppi
test-product-1	Test Product 1	0,50 EUR	0 %	5,00 EUR	1	5,00 EUR	Tuote
test-product-2	Test Product 2	1,50 EUR	0 %	1,00 EUR	1	1,00 EUR	Tuote

Kuva 21. Maksun yksityiskohdat Bambora PayFormin kauppiasportaalin

Kauppiaspaneeli (Paytrail)

Paytrailin kauppiaspaneeli näyttää maksut-sivulla hakuehtojen mukaiset maksut. Lisää hakuehtoja saa 'hae maksun tiedolla'-painikkeella. Haun tulokset saa tulostettua CSV-raportiksi. (Kuva 22.)

Paytrail Kauppiaspaneeli
Asiakaspalvelu: 020 718 1820 (0,0835 €/puhelu + 0,3669 €/min)

EN SV Kirjautu ulos

Etusivu Maksut Tilitykset Tilastot Viestit Asetukset

Maksut > Maksuarkisto

01.02.2018 07.02.2018 **Hae**

Maksutapa Kaikki Tila Onnistuneet, kaikki Palveluntarjoaja Kaikki

[Hae maksun tiedoilla](#) [Hae hyväksyntää odottavat laskut](#)

Maksujen määrä: 5 Summat yhteensä: 4,04 € Tilitykset yhteensä: 1,89 € Provisiot yhteensä: 1,45 € Keskimääräinen maksu: 0,81 € [Lataa CSV-raportti](#)

Aika	Tilausnumero	Tilaaja	Maksutapa	Tila	Summa	Tilittään	Tilityspäivä
06.02.2018 07:32:57	ORD-36158	Sauvamäki, Juha	Paytrail, Paypal	Maksettu	0,70 €	0,00 €	
06.02.2018 06:37:01	ORD-66100	Sauvamäki, J	Paytrail-tili, Visa	Maksettu (Siirretty tilitykseen)	0,70 €	0,34 €	13.02.2018
06.02.2018 06:31:51	ORDNETTI555	Sauvamäki, Juha	Paytrail-tili, Visa	Maksettu (Siirretty tilitykseen)	1,24 €	0,87 €	13.02.2018
06.02.2018 05:25:13	ORD-91479	Sauvamäki, Juha	Paytrail, Visa	Maksettu (Siirretty tilitykseen)	0,70 €	0,34 €	13.02.2018

Kuva 22. Paytrailin kauppiaspaneelin maksut sivu

Kauppiaan Extranet (Checkout)

Checkout on nimennyt palvelunsa Kauppiaan Extranetiksi. Kuvassa 23 näkyy maksutapahtumien hakuvalitsemat. Hakuvalitsemat ovat monipuoliset ja tulokset saa ladattua CSV-tiedostoon.

Maksutapahtumat

Etunimi Sukunimi Pvm. -

Summa - € Viite

Arkistointitunnus Kuvaus Tila

Maksutapa Tilitysraportti Palautukset

Hae [Piilota hakuvalitsemat](#) [Tyhjennä hakuvalitsemat](#) [Tallenna hakukriteerit](#)

Löytyi 1 tapahtuma [Lataa CSV](#)

Pvm.	Aika	Summa	Tila	Nimi	Maksutapa	Viite	Kuvaus
24.09.2019	13:51	30,70 €	Maksettu	Mari Testaaja	Nordea	245016255	Pikanouto tunnissa YA:n toi...

Kuva 23. Checkoutin Kauppiaan Extranetin maksutapahtumien hakuvalitsemat

Yhteenveto maksutapahtumien käsittelyn sovelluksista

Esitetyillä ohjelmilla voidaan seurata tilitapahtumia, tulostaa raportteja ja tehdä palautuksia. Yrittäjällä on myös velvoitteita, joiden hoitamiseen raporteista saa apua. Verolaki velvoittaa hoitamaan kuukausittain ALV ja ennakovero -tilitykset ja tilinpitolaki hoitamaan yrityksen kirjanpidon vuositason. Esimerkiksi Checkoutissa tilitysraportit saa kauppiaan extranetistä, ja ne voi myös asettaa menemään automaattisesti haluamaan sähköpostiin, vaikka kirjanpitäjälle.

Eri toimijoiden sovellukset ovat ominaisuuksiltaan hyvin yhtenevät. Maksujen palautukset voi hyvin tehdä täällä. Kaikilla saa rajattua hakutuloksia ja tulokset saa myös CSV-tiedostoon. Kaikki toimittavat ilmaisen ohjelman yhdelle käyttäjätunnukselle. Lisäkäyttäjistä maksetaan suhteellisen kallista kuukausihintaa. Eri käyttäjätunnuksille voi määrittää erilaiset oikeudet tapahtumien käsittelyyn, mikä voi olla järkevää isommissa verkkokauppojen organisaatioissa. Bamboralla tulee pientä lisähintaa kehittyneemmästä palautuksen hallinnasta.

Yhdenkään toimijan nettisivuilla ei tuoda näkyvästi esille näiden ohjelmien esittelyä eikä dokumentaatiota, vaikka ohjelmat ovat käytännössä verkkokaupalle pakollisia työkaluja. Minkäänlaista dokumentaatiota näistä sovelluksista ei edes löydy Paytraililta eikä Checkoutilta. Bamboran ohjelmaa voi testata testitunnuksilla ja nettisivuilta löytyi myös ohjeita ohjelmiston käyttöön. Kun asiakkaat tekevät sopimuksen, niin toimijat antavat käyttöoikeuden ohjelmistoihin ja toimittavat ohjeita niiden käyttöön. Jos kyseessä on uusi verkkokauppa, ilman mitään kokemusta maksupalveluista, tässä asiassa voi mennä metsään. API-dokumentaatioissa kerrotaan, kuinka voi tehdä palautuksia ja tulostaa raportteja. Tämän perusteella voi suunnitella oman käyttöliittymän, mikä tekee aivan samoja asioita kuin maksupalveluiden ilmaiset ohjelmistot.

3.4 Maksupalveluiden vertailun tulokset

Maksutapojen vertailulla saatiin karsittua maksupalveluita. Hintojen ja kauppaspaneelien vertailu eivät tuottaneet lisäkarsintaa toimijoille. Siksi valitaan tarkempaan tarkasteluun ja testailuun kaikki kolme jäljelle jäänyttä maksupalvelua. Kuvasta 24 nähdään, että karsituiksi tulivat BlueCommerce, Mash, Klarna, PayPal ja Stripe.

	Korttimaksu	Verkkopankki	Lasku tai osamaksu	Mobiilimaksu	PayPal tai vastaava	Toistuvat maksut
	✓	✓	✓	✓	✗	✓
	✓	✓	✓	✓	✓	✗
	✓	✓	✓	✓	✗	✗
	✗	✗	✓	✗	✗	✗
	✓	✓	✓	✗	✗	✗
	✓	✗	✗	✗	✓	✓
	✓	✓	✓	✓	✓	✗
	✓	✗	✗	✓	✓	✓

Kuva 24. Taulukko karsitutuista maksupalveluista

Karsinnan tekemistä voidaan pitää perusteltuna, koska se vähentää aikaa vievien testien tekemistä ja laajojen API-dokumenttien lukemista. Toisaalta verkkokaupan asiakkaille tärkeintä ovat monipuoliset maksutavat, eikä asiakas juuri muita eroja havaitse maksupalveluissa. Muut erot näkyvät vain verkkokaupalle: kauppiasportaaleissa tehtävissä toimenpiteissä, kuluissa sekä verkkokaupan sovelluskehityksessä.

4 VALITTUJEN MAKSUPALVELUIDEN ESITTELY JA TESTIT

Tässä luvussa tutkitaan tarkemmin kolmea maksupalvelua, jotka ovat karsinnan jälkeen valikoituneet parhaiksi vaihtoehdoiksi Mindhive Oy:n käyttöön. Maksupalvelut ovat Paytrail, Checkout ja Bambora PayForm. Toimijat esitellään ja kuvataan niiden palveluiden ominaisuudet ja lisäpalvelut. Tutustutaan palveluiden Form- ja API-rajapinta dokumentteihin ja toteutetaan niiden perusteella testejä. Yritysten esittelyjärjestys on sama, kuin missä järjestyksessä API-testit on tehty. Tämän luvun lopussa on maksupalvelun valinta. Siinä kootaan eri toimijoiden tiedot taulukoihin ja vertaillaan niitä. Sen vuoksi seuraavissa yritysten esittelyssä keskitytään kertomaan yrityksistä, niiden palvelusta yleisellä tasolla ja API-rajapinnasta.

4.1 Paytrail

Paytrail on suomalainen, Jyväskylässä sijaitseva maksupalveluyritys. Se on perustettu vuonna 2007 ja kuuluu Nets-konserniin, joka on iso toimija Pohjoismaissa. Paytrailin nettisivujen mukaan, Paytrail on maksuvolyymiltaan suurin

kotimainen verkkomaksupalvelu (Paytrail 2020). Paytrail vaikuttaa keskittyneen Verkkokauppa-alustojen liittämiseen. Se on valmiiksi integroitu lukuisiin kauppa-alustoihin ja osaan on saatavilla lisäosa, jolla se voidaan liittää.

Paytrail tarjoaa maksupalvelua vain yhdellä sopimuksella. Siinä on kiinteä kuukausihinta ja tapahtumakohtaiset maksut. Lisäksi osassa maksutapoja on provisiomaksut. Lisäpalveluna myydään myös maksuvalintasivun ohitus, mikä on muissa esitettävissä palveluissa ilmainen. Jos tarvitsee sen toiminnon, nousee tämä palvelun kiinteä kuukausikulu kaksi kertaa kalliimmaksi kuin vertailun halvimmassa palvelussa. Toisaalta vuosikulu ei ole kuitenkaan kuin 936 € / vuodessa, mikä on pieni kulu verrattuna suuren verkkokaupan liikevaihtoon. Paytrail on kirjoitushetkellä hivenen muita kalliimpi palvelu, mutta heidän sivuillaan kehoitetaan ottamaan yhteyttä ja pyytämään tarjousta. Yksi lisäpalvelu on rinnakkaistunnukset, jolloin eri henkilöille voidaan antaa eri oikeudet ohjelmiston käyttöön. Paytraililla on lisäpalveluna myös automaattinen Sales Rescue-palvelu. Palvelu lähettää asiakkaalle sähköpostia, jos asiakkaan tilaus on keskeytynyt maksupalvelussa. Sähköpostissa on linkki, jota kautta pääsee maksamaan kesken jääneen tilauksen. Toteutettavassa verkkokauppa-sovelluksessa tehdään siten, että asiakas palaa keskeytyneen maksun jälkeen suoraan verkkokauppaan, haetaan tilauksen tiedot ja asiakas voi yrittää maksua uudelleen. Sales Rescue toimii tätä tapaa paremmin, jos maksun keskeytyksen syynä on verkkoyhteyden katkeaminen tai nettiselaimen sulkeminen. (Lintinen 2020.)

Toteutustavat

Paytrailiin saa verkkokauppa-moduuleja, Form-rajapinnan, REST API -rajapinnan, mutta ei mobiili-API kirjastoa.

Form interface versio E2

Form interface on helpoin tapa tehdä maksutestejä. Testi onnistuu kopioimalla dokumentaation Form-esimerkin HTML bodyn sisälle ja käynnistää selaimella HTML-sivun. Klikkaamalla painiketta pääsee tekemään testimaksua Paytrailin sivun kautta eri maksutapoihin. Input-kenttien tietojen muokkaus johtaa virheeseen sähköisessä allekirjoituksessa. Dokumentaatiossa kerrotaan hyvin, miten tämä AUTHCODE lasketaan. Laskennasta löytyy esimerkki luvusta 2.4

kuvasta 9. Kun allekirjoitukselle on laskenta, voi helposti tehdä ohjelmaversi-
on, missä pääsee syöttämään maksutapahtuman tiedot. Form-rajapinnassa
voi myös tehdä maksun tilan kyselyn.

REST Interface (API)

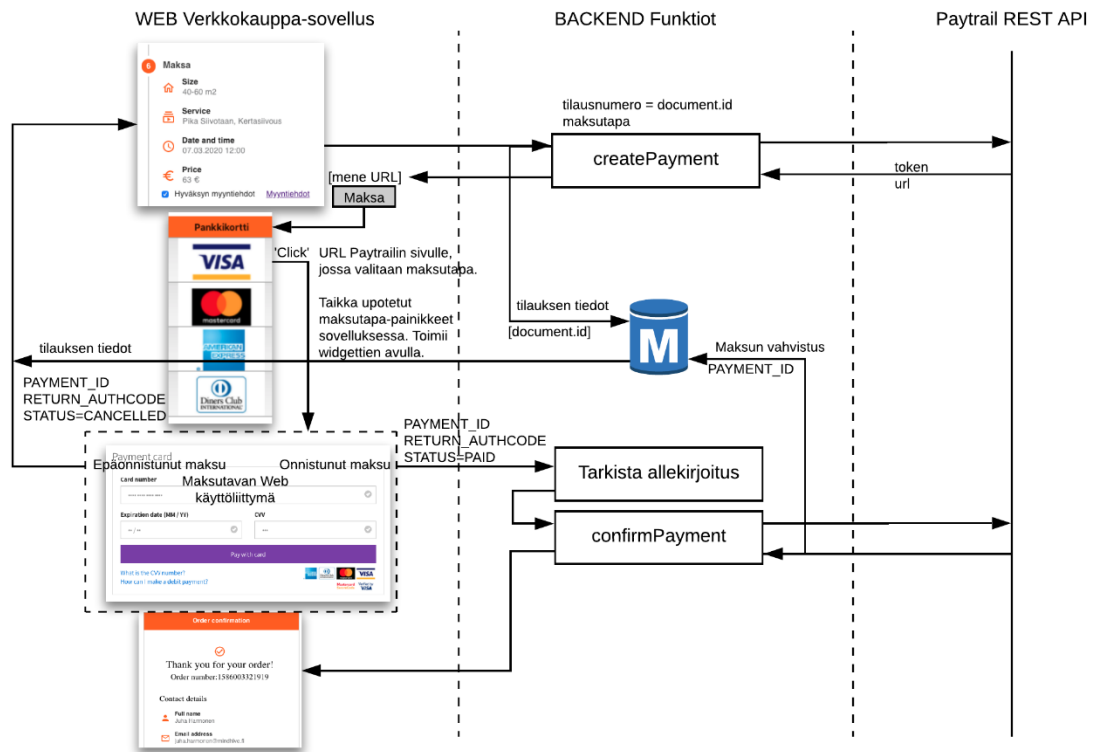
Tällä REST API -rajapinnalla voi lähettää maksun tiedot. Palautteena tulee
mm. linkki, jolla pääsee maksamaan kyseisen maksun. Maksun maksamisen
jälkeen, käyttäjä ohjataan takaisin verkkokauppaan tai muuhun haluttuun
osoitteeseen ja näytetään URL-kentässä maksun onnistumisen tiedot. Tämän
API-rajapinnassa voi tehdä myös maksun tilan kyselyn ja maksupainikkeiden
upotuksen verkkokauppaan. Paytraililta puuttuu pankkikorttien tallennustoi-
minnot.

Dokumentaatioissa (Paytrail 2019) löytyi hivenen PHP-mallikoodia maksun
suoritukseen ja sivulta saa ladattua API:n tehdyn PHP-luokan (Paytrail 2014).
Node.js koodia sen sijaan ei löydy dokumentaation kautta. GitHubista löytyy
aika niukasti tarjontaa Paytrailin REST API:lle. Kokeilin testissä yhtä sieltä löy-
tynyttä, Node.js koodilla tehtyä esimerkkiä (Vakula 2017). Sillä onnistui maksu
teko ja onnistuneen maksun tietojen allekirjoituksen varmistus. Tästä demosta
jäi puuttumaan maksun tilan kysely.

Merchant API v1

Tämä on erillinen API-rajapinta palautusten ja tilitysten hallintaan. Täällä voi
tehdä palautuksen, perua palautuksen ja tehdä kyselyjä verkkokaupan tilityk-
sistä halutulta aikaväliltä. Kyselyillä voi saada lisäksi yksityiskohtaisia tietoja
yksittäisistä tilityksistä, maksuista ja palautuksista. Yksi syy tämän rajapinnan
käyttöön voi olla se, että haluaa tehdä automaattisen tietojen siirron eri ohjel-
misten välillä. Palautukset ja maksutapahtumien tietojen tulokset saa
myös Paytrailin kauppiaspanelista. Tähän rajapintaan ei löytynyt demokoodia.
Testin tekeminen ei onnistunut pelkästään monimutkaisen dokumentaation
perusteella.

Kaikki edellä esitellyt toteutustavat ovat kuvattu yhdessä pitkässä dokumen-
tissa. Sen ulkoasu on vanhentunut ja siinä ei ollut sivussa sisällysluettelo.
Dokumentti kaipaa lisää koodi- ja viesti esimerkkejä.



Kuva 25. Paytrail REST API:n käyttö verkkokauppa-sovelluksessa

Kuvassa 25 esitetään, kuinka verkkokauppa sovelluksen front (web) tekee kutsuja backend funktioille, jotka ovat taas yhteydessä Paytrailin REST API:n. Backend välittää API:n vastauksen tiedot fronttiin, jossa ne esitetään verkkokaupan käyttäjälle. Kaaviossa, web näkymä päivittyy vaiheittain ylhäältä alas, käyttäjän toimien mukaan. Ensimmäisessä vaiheessa käyttäjä on edennyt verkkokaupassa maksusivulle ja sivun lataaminen käynnistää createPayment-funktion, tilauksen tietojen lähetyksen ja tallennuksen. Palautteena saadaan token ja URL-osoite Paytrailin maksuvalintasivulle. Tämä on helppo toteutettava ratkaisu. Verkkokauppaan laitetaan maksupainike, joka vie maksutavan valintaan Paytrailin sivulla. Jos haluaa tämän välivaiheen pois, pitää käyttää heidän toimittamaa widgettiä, jolla haetaan maksu painikkeet ja saadaan ne upotettua verkkosivulle. Widgetti tarvitsee jQueryn toimiakseen ja dokumentissa neuvotaan sen asennuksesta HTML-sivulle. Tässä toteutettavassa projektissa käytetään Reactia ja siinä voitaisiin myös käyttää jQueryä. Dokumentaatiossa ole kuitenkaan yhtään esimerkkiä, joten tämä jäi kokeilematta. Widgetissä säädetään parametreja, jolla vaikutetaan mm. painikkeiden koon. Painikkeen painaminen lähettää form-elementin ja käyttäjä ohjautuu maksutavan palveluun. Maksun jälkeen käyttäjä palautuu verkkokauppaan ja maksun tiedot tulevat URL-kenttään. Jos URL-kentän tiedot kertovat, että

maksu onnistui, tehdään vielä varmistava kysely maksupalvelusta, missä ehtona on, että sähköinen allekirjoitus on oikein. Tallennetaan palautteena saadut tiedot, erityisesti transactionId ja maksun tila. Jos maksu on suoritettu, näytetään tilausvahvistus käyttäjälle. Keskeytetystä tai epäonnistuneesta maksutapahtumasta käyttäjä palautetaan alkuun. Maksusivulle haetaan kesken jääneen tilauksen tiedot ja lähetetään uusi createPayment kutsu. Maksutapahtuman tiedot täytyy lähettää uudestaan, palautteena oleva token toimii vain yhden kerran.

4.2 Checkout

Checkout on suomalainen ohjelmistotalo, joka kuuluu OP Ryhmään. Checkout tarjoaa maksupalvelua eri toteutustapoihin: moduuleja verkkokauppa-aloille, verkko-API ja mobiili-API -rajapinnat. Tässä luvussa tutustutaan Checkoutin verkko-API:iin.

Checkout tarjoaa kolmea erilaista sopimusta. Startti on sopiva pieneen (alle 1500 € / kk) verkkokauppa käyttöön. Siinä kuukausimaksu on pieni, mutta tapahtumamaksut ovat hivenen kalliimpia ja rahat tilitetään vain kerran kuukaudessa. Laajassa sopimuksessa on suurempi kuukausimaksu, mutta tapahtumahinnat ja provisiot ovat kilpailukykyiset. Varat tilitetään viisi kertaa viikossa, 2-3 pankkipäivän sisällä maksun suorituksesta. Pro sopimus on kallein kuukausimaksultaan ja tarjoaa mahdollisuuden kauppiaan omiin korttitilityssopimuksiin Visan ja MasterCardin kanssa. Tilitykset saa siinä nopeimmillaan yhdessä pankkipäivässä. Missään sopimuksissa ei ole aloitusmaksua ja sopimusten vaihto onnistuu milloin tahansa, kun vain ottaa yhteyttä asiakaspalveluun. Checkout on kirjoitushetkellä ainut esiteltävistä palveluista, johon on tulossa Apple Pay ja Google Pay maksutavat. (myynti@checkout.fi 2020.)

Toteutustavat

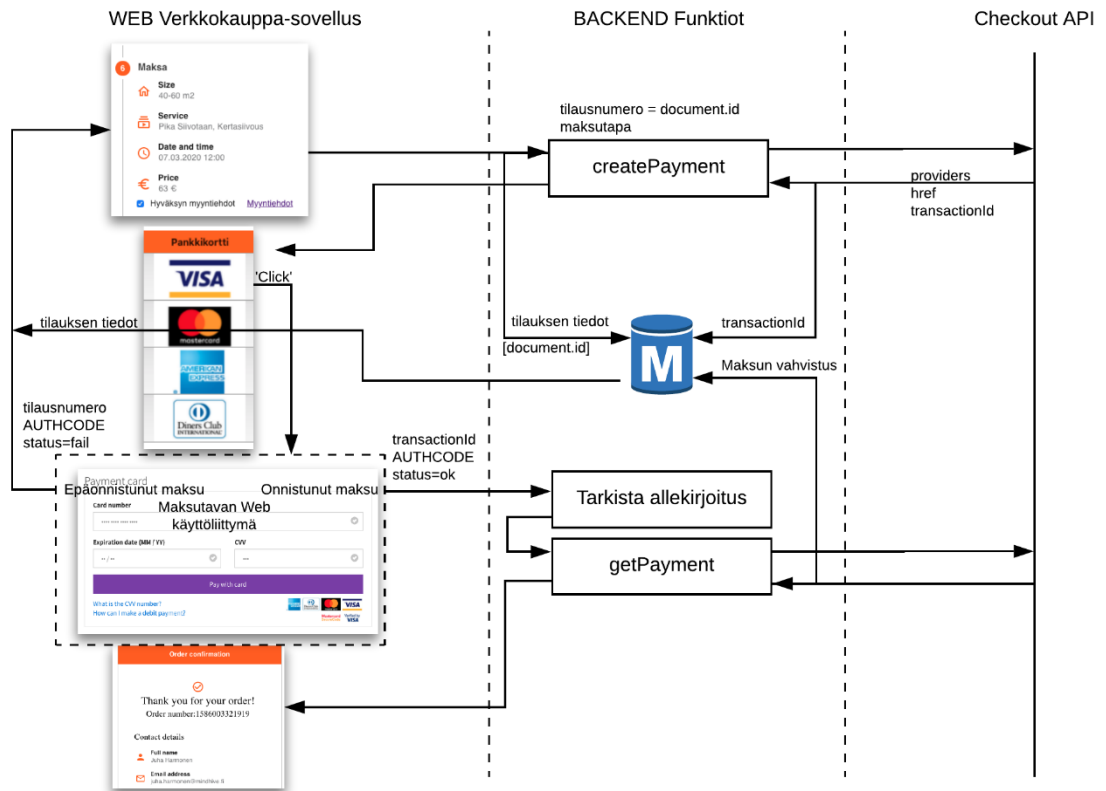
Checkoutissa on liitännät kaikkiin toteutuksiin: Verkkokauppa-moduuli, Verkko-API ja mobiili-API.

Verkko-API:n toiminnot

Uuden maksun muodostus ja kysely maksutapahtuman tiedoista. Maksun palautus osittain tai kokonaan. Luottokortin tallentaminen järjestelmään käyttäjän halutessa. Kortin tallennus helpottaa tulevia ostoksia, kun tietoja ei tarvitse syöttää uudestaan. Tallennus tehdään onnistuneen luottokorttiosoksen jälkeen. Vaikka suomalaiset käyttäjät tallentavat korttinsa harvemmin kuin muissa Pohjoismaissa (Verkkokauppa Suomessa 2019), pidetään tätä tapaa tietoturvallisena. API:sta löytyy Merchants-kysely, joka palauttaa listan saatavilla olevista maksutavoista. Tämän avulla verkkokauppaan voi upottaa maksutavan valinnan painikkeet.

Dokumentaation vasemmassa reunassa on sisällysluettelon linkit (Checkout Finland 2019). Sivua selatessa linkit näyttävät alaotsikoita. Linkit eivät kuitenkaan näytä missä vaiheessa dokumenttia käyttäjä on. Dokumentaatio on jaettu järkevästi muutamiin välilehtiin. Dokumentaatiossa on paljon linkkejä, jotka vievät dokumentin eri kohtiin. Ongelmana oli se, että selaimen paluupainike toi takaisin oikealle sivulle, mutta valitettavasti aivan sen alkuun. Dokumenttia lukiessa saman kohdan löytäminen oli linkin painamisen jälkeen vaikeaa. Dokumentaatio ja esimerkit ovat kuitenkin hyviä.

Checkoutin dokumentaatiossa ei ole viittauksia GitHubiin ja sen demokodeihin. Checkout Finland Oy:n viralliselta GitHubin sivulta löytyy jotain PHP-koodia, jota voi hyödyntää, mutta Node.js koodia ei löydy. Nettihauilla löytyy kattava Node.js demokoodi, jonka avulla saa testattua lähes kaikki API:n toiminnot (Nuurtamo 2019). Tämän demon funktioita käyttämällä saa tehtyä verkkokauppa-sovelluksen. Pienellä muokkauksella saa Node.js serverille tehdyt funktiot modifioitua Next.js ympäristön käyttöön.



Kuva 26. Checkout API:n käyttö verkkokauppa-sovelluksessa

Kuvasta 26 nähdään, kuinka ensimmäisessä vaiheessa käyttäjä on edennyt verkkokaupassa maksusivulle ja sivun lataaminen käynnistää tilauksen tietojen tallennuksen. Funktio lukee tallennettavan uniikin tiedoston nimen, jonka Googlen Firebase muodostaa. Tiedoston nimestä tehdään tilausnumero ja se lähetetään tilauksen tietojen mukana Checkout API:lle. Palautteena saadaan maksutapojen tiedot, kuten maksupainikkeiden kuvakkeet ja URL-osoitteet palveluun. Checkoutin dokumentissa on ohje, kuinka eri painikkeet laitetaan form-elementin sisälle kaikkine parametreineen. Painikkeen painaminen lähettää form-elementin ja käyttäjä ohjautuu maksutavan palveluun. Maksun jälkeen käyttäjä palautuu verkkokauppaan ja maksun tiedot tulevat URL-kenttään. Jos URL-kentän tiedot kertovat, että maksu onnistui ja allekirjoitus on oikein, tehdään vielä varmistava kysely maksupalvelusta. Tallennetaan palautteena saadut tiedot, erityisesti transactionId ja maksun tila. Jos maksu on suoritettu, näytetään tilausvahvistus käyttäjälle. Jos maksu epäonnistuu (status=fail), palautetaan käyttäjä kaaviossa alkuun ja näytetään tilauksen tallennetut tiedot.

4.3 Bambora PayForm

Bambora AB on iso kansainvälinen toimija, joka on laajentunut ostamalla useita maksamiseen erikoistuneita yrityksiä eri puolilta maailmaa. Suomalainen Paybyway Oy on yksi näistä toimijoista, kauppa tehtiin vuonna 2015. Vanha tuotenimi oli Maksukaista ja siihen kehitetty teknologia on nyt käytössä Bambora PayFormin nimellä. Tuotetta kehitetään edelleen Lappeenrannassa.

Sopimus vaihtoehtoja on kaksi, PayForm Express ja PayForm Power. Express on kuukausimaksuton ja Power kk-maksullinen. Eroja on käytännössä vain provisiomaksuissa. Pienen verkkokaupan toteutukseen Express on oikea vaihtoehto. Molemmilla sopimuksilla ovat maksuhyvitykset ja maksuvalintasi-
vun ohitus ilmaisia. (Kuva 27.)

PayForm Express		PayForm Power	
Ei kiinteää kuukausihintaa tai avausmaksua.		Ei avausmaksua. Sopimus voimassa toistaiseksi	
Kuukausihinta	0€	Kuukausihinta	39€
Pankkimaksut (kaikki)	0,35€ + 2,8 %	Pankkimaksut	0,35€
Korttimaksut	0,35€ + 2,8 %	Pankkimaksut (Nordea)	0,50€
Lompakot (MobilePay, Masterpas...)	0,35€ + 2,8 %	Korttimaksut	0,35€ + 1,9%
Lasku ja osamaksu (Jousto)	0,35€ + 2,8 %	Lompakot (MobilePay, Masterpass ...)	0,35€ + 1,9%
Maksuhyvitykset	0€	Lasku ja osamaksu (Jousto)	0,35€
Maksuvalintasi- vun ohitus	0€	Maksuhyvitykset	0€
		Maksuvalintasi- vun ohitus	0€

Kuva 27. Bambora PayFormin hinnoittelu 11.4.2020

Lisämaksusta voi ostaa eri palveluja (kuva 28). Kehittynyt maksupalautusten hallinta, eroaa ilmaisesta versiosta siten, että kaikilla maksutavoilla tehdyt ostokset voidaan palauttaa osittain, kokonaan, tuoterivikohtaisesti sekä tehdä monta palautusta yhteen maksuun. Lisäksi käytössä on PayForm-palvelun hyvityspyyntölomake, jolla asiakas voi pyytää palautusta kauppiaalta. Lomakkeen linkki on PayFormin sähköpostitse asiakkaalle lähettämässä maksuvahvistuksessa. Täytetystä lomakkeesta tulee ilmoitus kauppiaille ja sen hyväksynnän kautta voidaan tehdä sopiva palautus. Rinnakkaisilla kirjautumistunnuksilla voidaan eri käyttäjille asettaa täydet tai lukuoikeudet Bamboran kauppiaasportaaliiin. Bruttotilitys tarkoittaa, että tilitykset tilitetään täysimääräisenä ja Bamboran maksutapahtumista lähetetään erillinen lasku. Ilman lisäpalvelua maksukommissiot vähennetään tilityksistä. Erillinen korttilityssopimus on

ajankohtainen, jos verkkokaupassa tehdään paljon tai suurivolyymista kauppaa maksukorteilla. Bamboran erillisellä suoralla korttitilityssopimuksella tulee kuukausimaksu, mutta päästään edullisempaan maksutapahtumien hinnoitteluun korttimaksuissa. Lisäksi korttimaksuja voidaan vastaanottaa Amex (American Express) ja Diners Club maksukorteilta. (Tainio 2020.)

Valinnaiset lisäpalvelut	
Kehittynyt maksupalautusten hallinta	10€/kk
Rinnakkaiset kirjautumistunnukset	10€/kk/kpl
Bruttotilitys	20€/kk
Erillinen korttitilityssopimus	40€/kk + 0,35€/kpl

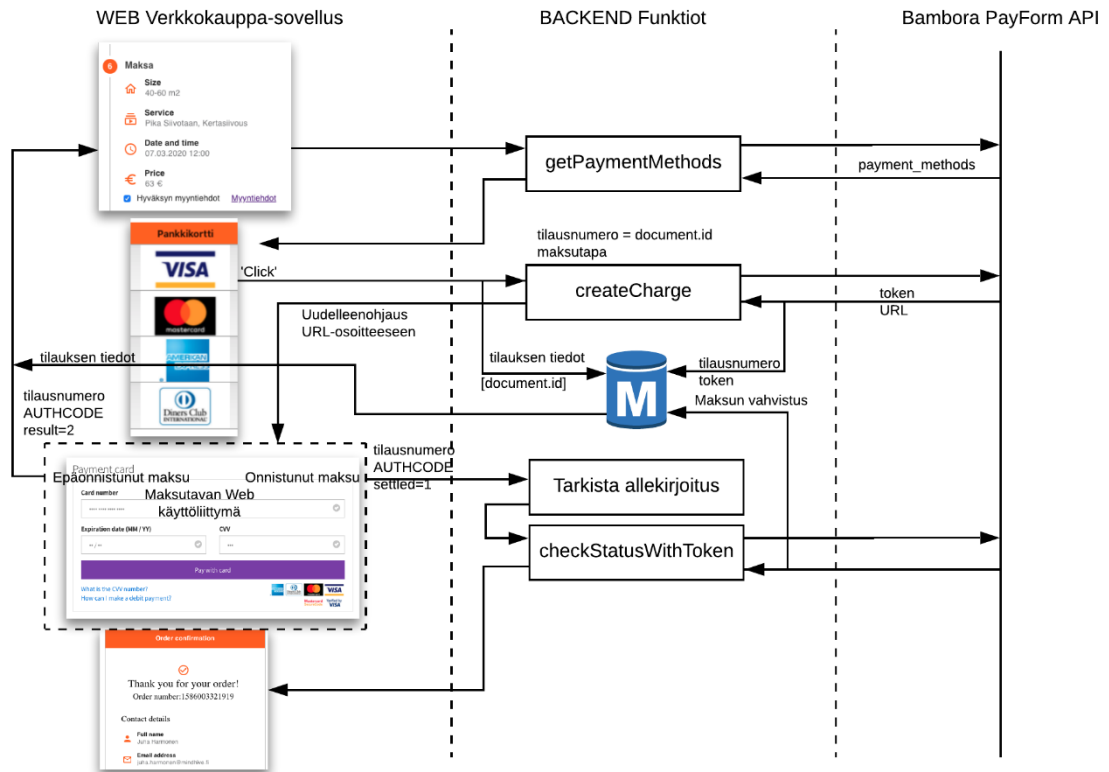


Kuva 28. Bambora PayFormin valinnaiset lisäpalvelut 11.4.2020

Toteutustavat

Bambora PayFormiin saa verkkokauppa-moduuleja ja REST API -rajapinnan. PayFormiin ei ole tarjolla SDK:ta, joten mobiilit sivut joutuu tekemään ilman nativejä näkymiä.

Web API:n Bambora tarjoaa ainoana toimijana sekä PHP-, että Node-kirjastot demoineen (Bambora 2019a). Kirjasto kattaa kaikki API:n toiminnot. Tämä on hyvä lähtökohta verkkokauppa-sovelluksen tekoa varten. Bamboran API:n logiikka eroaa jonkun verran Paytrailin ja Checkoutin vastaavasta. Bamborassa haetaan ensin erikseen maksupainikkeiden kuvakkeet funktiolla `getPaymentMethods` ja näytetään ne verkkokaupan käyttäjälle (kuva 29).



Kuva 29. Bamboran API:n käyttö verkkokauppa-sovelluksessa

Käyttäjän painamasta painikkeesta tunnistetaan valittu maksutapa, kirjoitetaan se maksun tietoihin, lisätään uniikki tilausnumero ja lähetetään maksun tiedot createCharge-funktiolla. Palautteena saadaan token, joka toimii maksutapahtuman tunnisteena ja osana URL-linkkiä. Tässä vaiheessa maksun kaikki tiedot, myös token ja tilausnumero, tulee tallentaa. Käyttäjä uudelleen ohjataan nettilinkkiin ja käyttäjä ohjautuu maksutavan palveluun. Maksun jälkeen käyttäjä palautuu verkkokauppaan ja maksun onnistumisen tiedot tulevat URL-kenttään. Maksun tunnisteena käytetään tilausnumeroa. Jos URL-kentän tiedot kertovat, että maksu onnistui, tehdään vielä varmistava kysely maksupalvelusta. Tarkastuksen ehtona on, että sähköinen allekirjoitus on oikein. Maksun ollessa maksettu, tallennetaan palautteena saatu tieto tilauksen tietoihin, merkitään tilaus maksetuksi ja näytetään tilausvahvistus käyttäjälle. Epäonnistunut maksutapahtuma (result=2) johtaa kaavion alkuun, josta tilausta voi jatkaa. Kaikki epäonnistuneet maksutapahtumat tekevät Firebasen tietokantaan ylimääräisen dokumentin, joita pitää myöhemmin siivota pois automaattisilla funktioilla.

4.4 Maksupalvelun valinta

Valinta tehdään vertailemalla maksupalveluja eri kriteereillä. Eri kriteereiksi valikoituivat maksuvaihtoehdot, laskulla maksaminen, kansainvälisyys, asiakaspalvelu, hinta, kauppiaspaneeli ja API-rajapinta. Usean eri kriteerin huomiointi tekee vertailusta monipuolisemman.

Maksuvaihtoehdot

Checkoutissa on eniten maksutapoja ja ne ovat kaikki käytössä ilman erillisiä sopimuksia yhdenkään maksuyrityksen kanssa. Checkouttiin on tulossa Apple Pay ja Google Pay maksutavat. Siinä on suurin valikoima mobiilimaksutapoja sekä Lasku ja osamaksu-maksutapoja. (Taulukko 2.)

Taulukko 2. Maksutapojen vertailu

	Bambora PayForm	Checkout	Paytrail
Korttimaksaminen	Visa, Visa Electron ja Mastercard.	Visa, Master-Card, American Express	Visa, Visa Electron, Mastercard, Eurocard, American Express
Mobiilimaksutavat	MobilePay, Masterpass ja Pivo.	Pivo, MobilePay, Masterpass, Siirto.	MobilePay, Siirto
Verkkopankit	Nordea, OP, Danske Bank, POP Pankki, Aktia, Säästöpankki, OmaSp, S-Pankki, Handelsbanken ja Ålandsbanken.	Nordea, OP, Danske Bank, POP Pankki, Aktia, Säästöpankki, OmaSp, S-Pankki, Handelsbanken ja Ålandsbanken.	Nordea, OP, Danske Bank, POP Pankki, Aktia, Säästöpankki, OmaSp, S-Pankki, Handelsbanken ja Ålandsbanken.
Lasku ja osamaksu	Jousto ja Enterpay-yrittyslasku.	Jousto, OP Osta laskulla, Collector, Mash, AfterPay	Jousto, Collector
Premium-maksutavat		Apple Pay, Google Pay	
Omalla sopimuksella	American Express ja Diners Club.		PayPal, Diners Club

Bambora PayForm ja Paytrail sisältävät myös kattavan määrän eri maksutapoja. Keskenään ne ovat aika samanveroisia, mutta muutamat erot voivat auttaa päätöksen teossa, riippuen siitä mitkä ovat tärkeitä maksutapoja yritykselle. PayFormiin saa Enterpay-yrittyslaskun ja Paytrailiin saa omalla sopimuksella PayPal-maksutavan. Mobiilimaksuissa on PayFormissa Masterpass ja Pivo, kun taas Paytrailissa on Siirto. Omia sopimuksia pitää tehdä, jos haluaa PayFormiin American Express ja Diners Club korttimaksut. Paytrailissa sopimukset tarvitaan PayPaliin ja Diners Clubiin. Checkoutissa puuttuu tässä mainituista maksutavoista PayPal, Diners Club ja Enterpay-yrittyslasku.

Laskulla maksaminen

Bambora PayFormiin on saatavissa Enterpayn järjestämä yrittyslaskutus ja sopimus tehdään suoraan Enterpayn kanssa. Enterpay laskuttaa kauppiasta kuukausittain, jokaisesta tämän maksutavan ostoksesta 1.6 % provision ja 5 €

laskutuslisän. Jos ostotapahtumia on vähän tai ei ollenkaan, laskutetaan palvelun minimikulu 19 €/kk. Kauppiaalla ei ole ostoksista luottoriskiä. Jos halutaan Enterpayn kulut siirtää asiakkaalle, se onnistuu muuttamalla ostoksen loppuhintaa valitun maksutavan mukaan.

Jousto-maksutapa tarjoaa laskulla maksamisen yksityisasiakkaille. Joustolla saa 30–3000 euron ostoksiin 30 vuorokautta korotonta maksuaikaa, ostokseen lisätään 2,90 € laskutuslisä. Kun asiakas saa laskun, voi hän valita myös luotollisen maksutavan ja pidentää maksuaikaa. Kauppias saa kuitenkin rahat tilausvahvistuksen jälkeen seuraavan viikon tilityksessä. Jouston kanssa voi tehdä joko suoran sopimuksen tai maksupalvelun kautta. Jouston käyttö on kauppiaille ilmaista ilman provisiota. Maksun palautus on ilmainen 45 vrk. laskun lähetyksestä. Jousto kantaa luottotappioriskin. Jousto on käytettävissä kaikissa kolmessa esitellyssä maksupalvelussa.

Kansainvälisyys

Verkkokauppa on helpoin keino päästä ulkomaiselle markkinalle. Ulkomaalaisille asiakkaille täytyy löytyä kuitenkin sopivat maksutavat. Eri maissa on eri tottumuksia maksutapojen käytöstä. Esimerkiksi muissa Pohjoismaissa, on korttimaksaminen kaikkein suosituin maksutapa ja tämä maksutapa toimiikin kaikissa palveluissa. Yhdelläkään toimijalla ei ollut tarjota näissä tuotteissa paikallisia maksutapoja. Bamboralla on ruotsalainen tuote Checkout, missä on maksutavat Ruotsiin ja se on paremmin linjassa myös muiden Pohjoismaiden tarpeisiin. Paytrailin emoyhtiön Netsin kautta on saatavilla laajempi maksuvalikoima Pohjoismaihin sekä muualle Eurooppaan. Näillä sopimuksilla ei ole kuitenkaan käytössä Suomalaisten verkkopankkimaksamista.

Bambora PayFormissa valitaan haluttu valuutta ja sille sopivat maksutavat palautetaan. Esimerkiksi Ruotsin kruunulle maksutapoina on luottokortit, Mobile-Pay ja Masterpass. Tilitysvaluuttana on kuitenkin aina euro. Paytraililla maksu ulkomailta onnistuu luottokorteilla ja Paypalilla, mutta valuuttana on aina eurot. Checkoutilla onnistuu luottokorttimaksut, mutta vain euroina.

Asiakaspalvelu

Tämän työn aikana on lähetetty kahteen kertaan kyselyjä sähköpostilla ja tuotu esille, että vastaukset on parasta käydä läpi puhelimesta. Mikään yrityksistä ei suoriutunut aivan puhtain paperein. Nopeimmalla vastausaika oli ensimmäisessä kyselyssä 2 päivää ja jälkimmäisissä kyselyissä yli viikon.

Paytrail oli nopein ja antoi hyviä vastauksia. Bamboran vasteaika oli hivenen pidempi, mutta vastaukset hoituivat erittäin hyvin. Checkout oli ainut, joka vastasi osaan kyselyistä nimettömästi asiakaspalvelun sähköpostiosoitteesta, jolloin jatkokysymysten esittäminen ei onnistunut. Heidän vasteaikansa oli toimituksista pisin, eikä sieltä myöskään tullut yhtään puhelua, jonka vuoksi varsinainen haastattelu jäi tekemättä. Kaikki asiat kuitenkin lopulta käsiteltiin, joten voidaan todeta, että kaikilla toimijoilla asiakaspalvelu toimii.

Hinta

Hintavertailun voitti PayForm kaikilla maksutavoilla. Siinä on pienimmät tapahtumamaksut ja provisiot ja lisäksi siinä oli pienin kuukausihinta. Checkout on hivenen kalliimpi ja Paytrail on jo reilusti kalliimpi, varsinkin kun huomioidaan maksusivun ohituksen kulu. Hint erot kalleimman ja halvimmän välillä ovat vuositasolla kuitenkin alle 500 €, jos verkkokaupassa on kuukaudessa yli 30 ostotapahtumaa. Aloittavalla verkkokaupalla saattaa olla ostotapahtumia vain satunnaisesti, jolloin Paytrail kiinteällä kuukausihinnalla on väärä valinta.

Kauppiaspaneeli

Bambora oli ainut, jonka maksutapahtumien käsittelyn sovellus toimi testitunnuksilla ja lisäksi ainut, josta löytyi edes dokumentaatio nettisivuilta. Bamboran ohjelma erottui positiivisesti myös ulkoasultaan. Ohjelmistojen toiminnoista ei löytynyt eroavuuksia, mutta näiden testit jäivät tekemättä Paytraililla ja Checkoutilla. Maksupalvelut voisivat panostaa näiden tarpeellisten ohjelmistojen esille tuomiseen ja dokumentaatioihin.

API-rajapinta

API-rajapinnat ovat maksun tarkistuksen osalta samanlaisia, ne eroavat vain parametrien määrässä ja niiden nimeämisessä. Parametrien esittely on myöhemmin taulukossa 3. API-rajapinnat eroavat toisistaan maksupainikkeiden haun osalta. Paytrailin API kutsuilla muodostetaan maksutapahtuma, johon

saadaan palautteena token ja URL-linkki. Paytrailin mahdollinen maksupainikkeiden upotus tehdään widgeteillä. Muutoin käytetään Paytrailin palvelua maksutavan valitsemisessa (URL-linkki). Checkoutissa muodostetaan maksutapahtuma, johon saadaan palautteena maksutapojen painikkeiden tiedot. Painikkeet sisältävät form-elementit ja niiden avulla käyttäjä ohjataan suoraan maksutavan palveluun. Bamborassa maksupainikkeiden haku on tehty erilliseksi API-toiminnoksi. Painikkeita käytetään vain maksutavan valinnan osoittamiseen, ne eivät sisällä maksukohtaista tunnistetta eikä form-elementtiä. Painikkeen painaminen kytketään maksutapahtuman API-funktioon, palautteena saadaan URL, jonka avulla käyttäjä uudelleenohjataan maksutavan palveluun.

Paytrailin REST API on hyvin niukka, siinä ei ole maksutavan painikkeille kutsua. Checkout ja Bambora PayForm sisältävät samat toiminnot, mutta eri tavoin toteutettuna. Checkout on hivenen yksinkertaisempi toteuttaa, siinä on yksi API-kutsu vähemmän. Toisaalta sen painikkeet sisältävät form-elementtejä lukuisine maksun parametreineen. Bambora PayForm on jaettu selkeisiin kokonaisuuksiin, maksupainikkeiden hakemiseen, maksun suorituksen aloitukseen ja maksun tarkistukseen. Bamboralla maksu alkaa vasta maksupainikkeen painamisesta ja mitään ei tallenneta ennen sitä. Tämä voi vähentää keskeytyneiden tilausten muodostumista tietokantaan.

Kaikkien dokumentaatiot olivat vain englanninkielisinä. Kyseessä on vakiintunut käytäntö teknisille dokumentaatioille. Nämä maksupalvelut ovat kuitenkin valtaosaltaan käytössä Suomessa ja suomalaisten tekemissä palvelussa. Paras dokumentaatio oli Bamboralla ja huonoin Paytraililla. Bambora tarjoaa esimerkillisesti sivuillaan linkkejä GitHubin PHP- ja Node-kirjastoihin ja esittelee sivuillaan niille tehtyjä omia dokumentteja, josta löytyy myös esimerkkikoodia. Muut toimijat eivät yllä lähellekään tätä. Checkout tarjoaa joitakin koodiesimerkkejä (PHP ja Node) sivuillaan ja Paytrail tarjoaa PHP-kirjastoa. GitHubista löytyy molempiin jotain, mutta ne eivät ole virallisia valmistajan tukemia kirjastoja.

Mobiili-API ja siihen liittyvät SDK-kirjastot löytyvät vain Checkouttiin. Sitä ei tässä työssä testattu, mutta kattava ja hyvä dokumentaatio siihen löytyy. Mo-

biililaitteilla tehdyn verkkokaupan osuus on Suomessa 39 % (Checkout Finland 2020), joten vaikka nyt ei olisi toteuttamassa verkkokauppaa mobiililaitteille, niin kannattaa huomioida tämä asia. SDK-kirjaston hyöty on se, että se mahdollistaa natiivit näkymät. Avointa API-rajapintaa voi kuitenkin käyttää mobiilisovellusten tekemisen, myös muihin maksupalveluiden palveluihin. Näkymät vain täytyy tehdä ilman tätä kirjastoa.

Kun palataan verkkokauppa-sovellukseen maksutapahtuman jälkeen, on maksupalveluissa käytössä samanlainen käytäntö: selaimen URL-kenttään kirjoitetaan tiedot maksutapahtumasta. Tätä käytäntöä tarvitaan, koska selaimella oleva verkkokauppa-sovellus latautuu uudestaan ja se unohtaa kaikki tietonsa siellä juuri tehdystä tilauksesta. Näiden tietojen ansiosta verkkokauppa palauttaa ostoksen tiedot ja jatkaa tilauksen hoitamista. Parametrit ovat eri nimisiä ja niitä on eri määrä. Taulukossa 3 on koottu eri toimijoiden parametrit samaan taulukkoon vertailua varten.

Taulukko 3. URL-parametrit, kun asiakas palaa verkkokauppaan maksutapahtuman jälkeen.

	Paytrail	Checkout	Bambora PayForm
Allekirjoitus	RETURN_AUTHCODE	signature	AUTHCODE
Maksun tunniste	PAYMENT_ID	checkout-transaction-id	[ORDER_NUMBER]
Palautteen tila	STATUS	checkout-status	RETURN_CODE
Maksun tila	[palautteen tilasta]	[palautteen tilasta]	SETTLED
Tilausnumero (Verkkokaupan)	ORDER_NUMBER	checkout-stamp	ORDER_NUMBER
Maksutapa	PAYMENT_METHOD	checkout-provider	
Maksun summa (cents)	AMOUNT	checkout-amount	
Aikaleima	TIMESTAMP		
Valuutta	CURRENCY		
Tunnistenumero oma tai Paytrail	SETTLEMENT_REFERENCE	checkout-reference	
Verkkokaupan ID		checkout-account	
Allekirjoituksen algoritmi		checkout-algorithm	

Taulukossa kehystettynä olevat tiedot ovat viestin merkitsevät tiedot, muut voidaan luokitella lisätiedoiksi. Sulkeissa olevat tiedot päätellään muista tiedoista, sille ei ole omaa parametria kyseisellä maksupalvelulla. Paytraililla, vihreällä värillä merkittyjen kenttien parametrit, tulevat vain niin halutessa. Vähemminkin tiedoilla pärjää, mutta Bambora tapauksessa maksun tunniste puuttuu parametreistä ja se aiheuttaa sen, että joko hakee sen tilaukselle tallennetusta token-tiedosta, tai luottaa siihen, että maksun tilausnumero on uniikki Bamboran tietokannassa. Itseasiassa Bamboran API on tehty niin, että kaikkien maksutapahtuminen tilausnumero täytyy olla uniikki, joten tämä ei ole

ongelma. Näiden kaikkien parametrien tiedot voidaan varmistaa, kun suoritetaan sähköisen allekirjoituksen uudelleenlaskenta. Onnistuneen maksutapah-tuman jälkeen voidaan tehdä varmistava maksun tilan kysely. Jos maksu on maksettu, tallennetaan se tilauksen tietokannan tietoihin ja näytetään asiak-kaalle tilausvahvistuksen tiedot. Nämä tarkistukset ja tallennukset tulee suorit-taa backendissä. Jos parametrit kertovat, että ostostapahtuma on epäonnistu-nut, haetaan tilausnumerolla tietokannasta tilauksen tiedot ja annetaan asiak-kaalle mahdollisuus yrittää maksun suoritusta uudestaan.

Yhteenveto tuloksista

Maksupalveluluiden valintakriteerien mukainen vertailu on nyt tehty ja tässä on sen yhteenveto. Jokainen voi painottaa näitä kriteereitä haluamallaan ta-valla. Osa arvioista on varsin subjektiivisia, mutta usean eri kriteerin huomioi-minen tekee vertailusta luotettavamman. Kriteerit on pisteytetty ja painotettu Mindhiven kannalta. Taulukossa 4 nähdään, miten maksutavat ja API nousivat vertailun tärkeimmiksi kriteereiksi.

Taulukko 4. Yhteenvedon laskentataulukko

PISTEYTYSTAULUKKO	PayForm	Checkout	Paytrail
Maksuvaihtoehdot	10	12	10
Laskulla maksaminen	2	1	1
Kansainvälisyys	2	1	1
Asiakaspalvelu	2	1	2
Hinta	5	4	2
Kauppiaspaneeli			
- Ominaisuudet	1	1	1
- Toiminnallisuus	2	1	1
- Toimii Demossa	1	0	0
API-rajapinta			
- Dokumentaatio	3	2	1
- Demot / kirjastot	3	1	0
- Mobiili-API	0	2	0
YHTEENSÄ	31	26	19

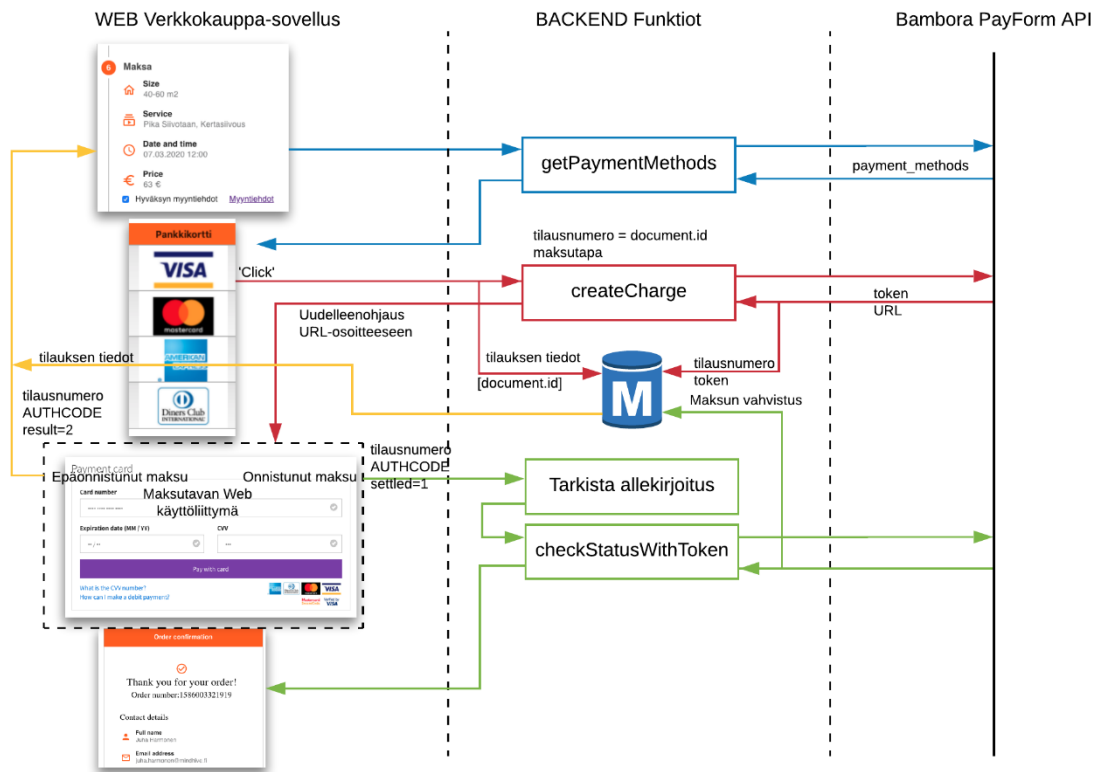
Maksuvaihtoehtoja on painotettu reilusti eniten, sen vaikutus eri toimijoilla on 32–52 % kokonaispisteistä. API rajapinnan osuus 5–19 %, hinnan osuus 11–16 % ja kauppiaspaneelin osuus 8–13 %. Loput kriteerit on arvosteltu asteikolla 0–2 pistettä, jolloin yhden kriteerin vaikutus on eri toimijoilla 0–11 %. Bambora PayForm on vertailun voittaja ja se pärjasi hyvin kaikilla kriteereillä.

5 VERKKOKAUPPASOVELUKSEN TOTEUTUS

Toteutettavaa verkkokauppa-sovellusta on esitelty jo aikaisemmin maksupalveluiden API-rajapinta kuvauksissa. Maksupalveluksi on valittu Bambora PayForm ja siten sovellus toteutetaan sen vaatimana versiona. Sovellus toteutetaan Web-sovelluksena käyttäen Next.js:ää ja backend hoidetaan Googlen Firebaseen funktioilla.

5.1 Sovelluksen tapahtumaketjut

Työssä on aikaisemmin perusteltu, miten koodi tulee jakaa kaikilta kriittisiltä osiltaan backend-funktioihin. Käytännössä se tarkoittaa, että kaikki API-kutsut, salauslaskenta, salaisten avainten käsittely ja tallennukset tehdään backendissä. Web sovellus näyttää käyttäjälle kaiken tarpeellisen ja välittää käyttäjän syöttämät tiedot backendin funktioille. Toisaalta kaikki toiminta tapahtuu Web-sovelluksen ohjaamana, backend tekee vain tarvittavat toimenpiteet ja antaa vastaukset sen tekemiin kyselyihin. Käydään seuraavaksi läpi nämä sovelluksen tapahtumaketjut.



Kuva 30. Verkkokauppa-sovelluksen eri toimintaketjut

Käyttäjää tulee maksu sivulle

Maksutavat tuodaan `getPaymentMethods` API-kutsulla ja painikkeet renderöidään käyttäjälle siten, että niiden painaminen käynnistää maksamisen ja parametrina on valittu maksutapa. Kuvan 30 kaaviossa, tämä kohta on merkitty sinisellä värillä.

Käyttäjää painaa maksutavan painiketta

Tilauksen tiedot tallennetaan dokumentiksi tietokantaan ja uniikki dokumentin nimi annetaan tilausnumeroksi. Tilausnumero ja valittu maksutapa tallennetaan maksutapahtuman tietoihin. Backend lähettää `createCharge` API-kutsulla maksutapahtuman tiedot. Backend palauttaa URL-osoitteen, mikä käynnistää Web-sovelluksessa uudelleenohjauksen kyseiseen osoitteeseen. Kaaviossa tämä on merkitty punaisella värillä.

Onnistunut maksu

Maksun jälkeen käyttäjä ohjautuu takaisin Web-sovellukseen ja URL-parametrit kertovat maksutapahtuman tiedot. Jos tiedot kertovat, että maksu on suoritettu, niin parametrit lähetetään backendille. Funktio tarkistaa tietojen sähköisen allekirjoituksen, tekee maksun tarkistuksen API-rajapinnasta ja tallentaa

tilaukseen tietoihin maksun vahvistuksen. Kaaviossa tämä on merkitty vihreällä värillä.

Epäonnistunut maksu

Maksutapahtuma epäonnistuu tai keskeytetään maksutavan palvelussa ja käyttäjä palautetaan verkkokauppaan. URL-parametrit kertovat, että maksutapahtuma on epäonnistunut. Tilauksen tiedot ladataan ja kaikki alkaa alusta. Käyttäjä ohjataan maksusivulle, jolloin vaihe 1. toistetaan ja myöhemmin myös muut vaiheet. Kerrotaan asiakkaalle, että maksutapahtuma keskeytyi ja että asiakas voi yrittää maksua uudelleen. Kaaviossa tämä on merkitty keltaisella värillä.

5.2 Bamboran API:n liittäminen Backend funktioihin

Bambora toimittaa Node.js-kirjaston ja sen /lib-kansiossa on payform.js-tiedosto (Bambora 2019b). Tätä tiedostoa käyttäen, saa API:n toiminnot käyttöön. Se kannattaa pitää erillisenä tiedostona ja tuoda se import-komennolla funktioiden käyttöön.

Kuvan 31 esimerkissä on funktio, jolla haetaan maksutapojen tiedot. Funktio on sama kuin kuvan 30 kaaviossa kuvattu getPaymentMethods.

```
import * as payForm './payform';

const getPaymentMethods = () => {
  return new Promise((resolve: any, reject: any) => {
    payForm.getMerchantPaymentMethods('', (error: any, currency: any, result: any) => {
      let response: any = '';
      if (error) {
        console.log(`Got error:${error.message}`);
        reject(error);
        return;
      } else if (result.result !== 0) {
        reject('Unable to get merchant payment methods');
        return;
      } else {
        response = result.payment_methods;
      }
      resolve(response);
    });
  });
};

.....

try {
  const merchantResult = await getPaymentMethods();
} catch (error) {
  console.log('Error in methods: ', error.message);
  throw error;
}
```

Kuva 31. Bamboran API:n käyttäminen backendissä

Bamboran payForm -rajapinta hyödyntää callback -funktioita palveluvastauksen käsittelyyn. Jotta rajapinnan hyödyntäminen olisi järjestelmänkehittäjille helppokäyttöinen, on toteutuksessa kutsut kapseloitu erilliseen `getPaymentMethods` -funktioon, jolla vastausten käsittely voidaan tehdä hyödyntämällä Promise-kutsuja. Funktiossa käydään mahdolliset virheet läpi ja muussa tapauksessa response-muuttuja saa arvoksi API:n palauttamien maksutapojen tiedot. Promise odottaa, että `resolve` palauttaa funktion onnistuneen lopputuloksen. Virheet erotellaan `reject`:illä, mikä palauttaa promicen hylkäyksen ja sille annettu arvo palautetaan `error`-objektiin (Javascript.info 2020). Loppujen lopuksi, tulos palautetaan (koodissa: `return new Promise`), `getPaymentMethods`-funktioita kutsuneelle koodille. Kutsuva koodi täytyy olla `try – catch` rakenteen sisällä, jotta `reject` tilanteet voidaan käsitellä. Tällä tavoin saa Bamboran API:n muutkin toiminnot käyttöön.

5.3 Palvelun varausten käsittely

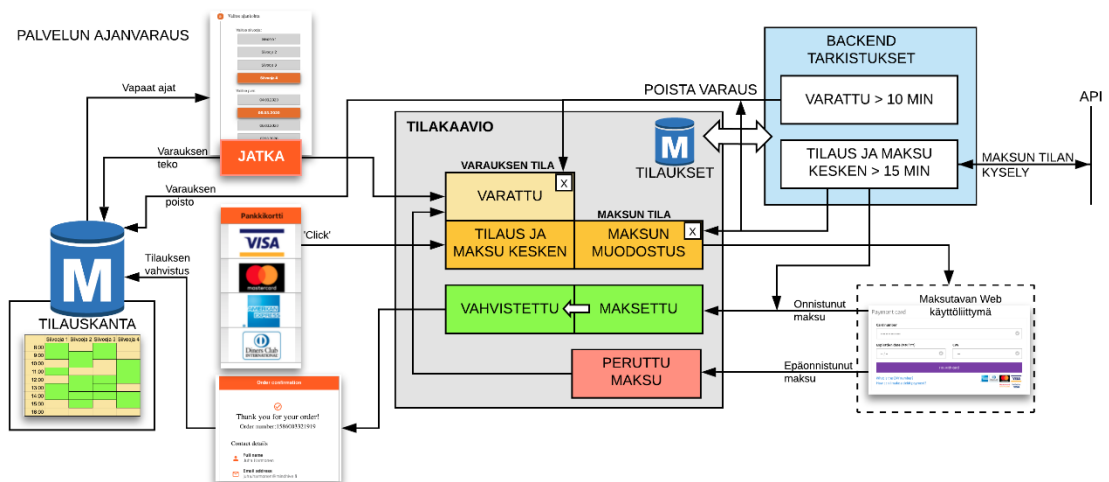
Tässä työssä toteutettavassa verkkokauppa-sovelluksessa myydään aikaan sidottuja palveluita. On tärkeää, että tilauskantaan ei jää ”haamutilauksia”, jotka ovat kesken jääneitä varauksia. Asiakas valitsee itselle sopivan ajan ja siirtyy maksamaan palvelun. Joskus asiakas saattaa jättää maksun suorituksen kesken, jolloin tallennettu varaus tulee poistaa tilauskannasta. Varausprosessin valvonnan voi mieltää kahdessa vaiheessa. Ensimmäinen vaihe on se, jos käyttäjä ei paina pankkipainiketta tietyssä ajassa. Toinen vaihe alkaa pankkipainikkeen painamisen jälkeen. Tästä hetkestä lasketaan lisäaikaa, jonka aikana maksu tulee olla suoritettu. Kun maksu menee normaalisti, niin asiakas palaa verkkokauppaan maksun jälkeen ja silloin maksu ja palvelun varaus voidaan vahvistaa heti. Hallittu tilanne on myös se, jos asiakas keskeyttää maksun maksutavan palvelussa ja palaa takaisin verkkokauppaan. Ensimmäinen ongelmatilanne on se, kun maksu jää kesken pankissa. Maksutapahtuma jää keskeneräisenä selaimella auki tai selain suljetaan. Tällöin se jää luonnollisesti myös maksamatta. Toinen ongelmatilanne on, kun asiakas suorittaa maksun, mutta sulkee selaimen ikkunan, ennen kuin käyttäjä uudelleenohjataan takaisin verkkokauppaan. Nämä kesken jääneet tilanteet tulee hoitaa backendillä suoritettavilla prosesseilla. Yksi merkitsevä asia on se, että kuinka pitkään varaus voi olla vahvistamatta. Tutkitaan seuraavaksi näitä tapauksia.

Taulukko 5. Varausten vahvistukseen tai perumiseen kuluva aika

TAPAHTUMA	VARAUKSEN TILA	MAKSUN TILA	SALLITTU AIKA TAPHTUMALLE	MAKSIMI AIKA VARAUksesta KULUNUT AIKA
Painettu Jatka-nappulaa varaus sivulla	Varattu	Ei aloitettu	0 - 10 min	10 min
Painettu pankkipainiketta	Tilaus ja maksu kesken	Maksun muodostus	0 min	10 min
Maksettu ja palattu verkkokauppaan	Vahvistettu	Maksettu	0 - 15 min	25 min
Peruttu maksu ja palattu verkkokauppaan	Varattu	Peruttu maksu	0 - 15 min	25 min
MAKSUTAPAHTUMAN KYSELYN VASTAUS Backend tarkistaa 10 min välein vahvistamattomia maksuja				
(Kesken) tai huoltokatko -->	Tilaus ja maksu kesken	Backend tarkistaa 1. kerran	0 - 10 min	35 min
Maksettu -->	Vahvistettu	Backend tarkistaa 2. kerran	10 min	45 min
Maksu epäonnistunut -->	Peruttu	Backend tarkistaa 2. kerran	10 min	45 min
		Backend tarkistaa 3. kerran	10 min	55 min

Taulukossa 5 kartoitetaan varauksen tietojen vahvistukseen kuluvaa maksimi aikaa. Taulukon ajat on saatu käyttäen muutaman oletuksen mukaisia alkuarvoja. Asiakkaalle annetaan maksimissaan 10 minuuttia aikaa painaa pankkipainiketta. Jos aika kuluu umpeen, palautetaan asiakas varauksen valintaan uudestaan. Maksutavan palvelussa maksun suoritus voi kestää maksimissaan 15 minuuttia. Tämä on testattu Osuuspankin palvelussa. Backendillä tehdyt maksutapahtumien tarkistusprosessit käynnistyvät 10 minuutin välein. Näillä ajoilla on laskettu kahden eri tapaukseen kuluvat maksimiajat. Niiden tulokset ovat oikeanpuoleisissa sarakkeissa kehystettyinä olevat 25 min ja 45 min.

Varauksen ja maksun tilat voi kyllä ymmärtää hyvin taulukkomuodossa, mutta on vaikeampi hahmottaa, mitä kaikkea tapahtuu niiden ympärillä. Kuvassa 32 on piirretty taulukon 5 mukainen tilakaavio, kuvassa se on harmaalla pohjalla. Sen tiedot kuuluvat Tilaukset-tietokannan tietoihin. Vasemmassa reunassa on palveluita myyvän yrityksen Tilaukset, jonne varauksen tilat päivitetään.



Kuva 32. Palvelun varauksen tilakaavio

Ensimmäinen tapaus alkaa alusta. Tilauskannasta haetaan vapaana olevat ajat, Asiakas valitsee palvelun ajan ja painaa Jatka-painiketta. Tästä alkaa ajan laskeminen (vanhenee 10 minuutissa) ja tilauksen tilaksi tulee Varattu. Kun asiakas painaa pankkipainiketta, tilauksen tilaksi tulee "Tilaus ja maksu kesken", jolloin varaus saa 15 minuuttia lisääaikaa. Ensimmäisessä tapauksessa maksutapahtuma saatetaan loppuun, joko maksun suorituksella tai keskeytyksellä. Tähän voi maksimissaan kulua 25 minuuttia varauksen tekemisestä. Käyttäjä palautuu verkkokauppaan, jolloin päivitetään tilauksen ja maksun tiedot. Jos maksu onnistui, niin maksu ja palvelun varaus vahvistetaan. Jos maksu epäonnistui, niin maksun tallennus voidaan poistaa ja palvelun varauksen tilaksi palautetaan ensimmäisen vaiheen "Varattu". Tällöin alkuperäisestä varaushetkestä kuluvan 10 minuutin voimassaolo palautuu. Jos maksun keskeytyksen jälkeen alkuperäinen 10 minuuttia on kulunut, ohjataan asiakas valitsemaan sopivaa aikaa, ennen kuin maksua voi jatkaa. Uudelleen valitun ajan jälkeen kaikki ajan laskeminen alkaa alusta.

Toisessa taulukon tapauksessa maksu on jäänyt kesken siten, että asiakas ei ole palannut verkkokauppaan. Backendin tarkistusprosessi hoitaa nämä tapaukset. Pankkipainikkeen painaminen muodostaa tallennetun maksutapahtuman dokumentin, joka sisältää mm. varauksen tiedot ja aikaleiman. Prosessi hakee kaikki yli 15 minuuttia vanhemmat maksutapahtumat ja kysyy niiden maksujen tilaa Bamboran API:sta. Vastauksena voi olla seuraavat tiedot: maksettu, maksu epäonnistunut, maksu kesken ja huoltotauko. Maksettu tieto johtaa maksun ja siihen liitetyn varauksen vahvistukseen. Toisaalta epäonnistunut maksu, poistaa maksun ja siihen liitetyn varauksen. Jos maksu on kesken tai palvelussa on huoltotauko, täytyy tätä kysyä seuraavalla kerralla uudelleen. Taulukon esimerkissä on huomioitu yksi mahdollinen ylimääräinen tarkastuskerta, jolloin maksimi kesto varauksen vahvistukselle olisi 45 minuuttia. Jos funktio menee vikatilaan tai huoltokatko jatkuu, voisi olla hyvä tehdä seuraavan tason varmistava tarkistus. Se voisi olla esimerkiksi tunnin välein ajettava hakutulos, joka näyttää yli tuntia vanhemmat vahvistamattomat maksut ja varaukset. Jos vahvistamattomia tietoja löytyy, voi tulokset lähettää ylläpitoon.

Backendin prosessin tulee käydä myös varauksia läpi. Kaikki varaukset, joiden tilaksi on jäänyt "Varattu" ja jonka ikä on yli 10 minuuttia, tulee poistaa tilauskannasta. Nämä varaukset syntyvät, kun asiakas valitsee ajankohdan ja

menee maksusivulle, mutta ei paina pankkipainiketta 10 minuutin kuluessa. Lisäksi keskeytetty maksutapahtuma muuttaa varauksen "Varattu"-tilaan.

6 PÄÄTÄNTÖ

Tässä työssä maksupalveluiden tutkiminen on tehty tutustumalla eri toimijoihin. Aluksi mietittiin, mitä toimijoita otetaan aikaa vieviin API-testeihin. Toimijoita oli kuitenkin useita ja niitä löytyi koko ajan lisää - oli selvää, että kaikkia ei voida testata, mutta ei ollut selvää, millä tavoin karsinta tehtäisiin. Tärkeässä osassa olivat tutkimukset, joista selvisi, että sopivat maksutavat ovat verkkokaupan asiakkaille kaikkein tärkeintä maksujen loppuun asti suorittamisessa. Tutkimuksissa vertailtiin myös eri maksutapojen yleisyyttä Suomessa sekä lähialueilla. Eri toimijoiden käytössä olevien maksutapojen vertailu helpottui, kun löytyi vuonna 2019 julkaistu blogi verkkokaupan maksutavoista. Siinä oli koottu samaan taulukkoon kahdeksan Suomessa käytössä olevaa maksupalvelua ja niiden maksutavat. Siinä oli mukana kaikki, joiden tutkintaa olin jo suorittanut ja lisäksi siinä oli muutama uusi. Tämä johti siihen, että työn tutkitavat maksupalvelut rajattiin näihin kahdeksaan maksupalveluun. Joidenkin toimijoiden palvelut jäivät huomioimatta tässä työssä ja se johtui tästä alkuvaiheesta tehdystä rajauksesta. Ainakin Svea on sellainen toimija, joka olisi voinut pärjätä testeissä. Tutkimusta voisi jatkokehittää, tekemällä toimijoille esikarsinnan siten, että maksupalveluista valitaan vertailuun mukaan vain ne, joissa on verkkopankkimaksaminen mahdollista ja joista löytyy avoin API-rajapinta.

Maksupalveluiden karsinnan jälkeen, työssä on testattu kolmen valitun maksupalvelun API-rajapintoja ja toteutettu verkkokauppa-sovellus. Testit aloitettiin Paytrailin kanssa ja lopetettiin Bamboralla. Checkoutin testien lopussa, olivat verkkokauppa sovelluksen perustoiminnot valmiit. Bamboralla oli täten paras asetelma testien tekemiseen, mutta se ei vaikuttanut tutkimustuloksiin. API-testit tehtiin yhdelle toimijalle kerrallaan ja testien aikana on opittu monta asiaa. Kokemuksesta olisi ollut ehkä hyötyä Paytrailin Merchant API v1:n testien toteuttamisessa, jotka eivät onnistuneet työn alussa. Tässä nähdään myös dokumentaatioiden ja demojen merkitys ainakin aloittelevalla ohjelmoijalle. Jos ohjelmoijalla on mahdollisuus vaikuttaa toimijan valintaan, niin nämä

asiat nousevat esille. Alussa tuntui, että API-dokumentit ovat hyvin vaikealukuisia, eikä ollut selvää, mitä toimintoja verkkomaksun suorittamiseen tarvitaan. Hyvin tekninen ja pitkä dokumentaatio ei auttanut kokonaisuuden hahmottamisessa. Toivon, että tämä työ auttaa kaavioiden avulla lukijaa tässä asiassa. Eri toimijoiden palveluista löytyy loppujen lopuksi samanlaisia API-toimintoja, niiden toteutus on vain vähän erilainen.

Työn tekemisen aikana on noussut esille muutamia asioita, jotka voisi nähdä kehitysehdotuksina maksupalveluille. Maksupalvelut haluavat uusia asiakkaita ja sen vuoksi heidän kannattaisi tarjota mahdollisimman helppo polku verkkokaupan tekemiseen myös API-rajapintaan. Nyt lähtökohtana toimijoilla on toimittaa lähinnä vain tekniset API-dokumentit, jossa on määritelty kaikki mahdolliset toiminnot. Toisaalta kaikki voisi lähteä selkeistä kuvista, mitä API-toimintoja verkkokaupassa tarvitaan eri toimijoilla. Tästä työstä löytyy esimerkit siihen luvusta 4. Seuraava vaihe voisi olla jo kauppiaspaneelin ominaisuuksien esittely. Kannattaa kertoa uusille asiakkaille, että palautukset ja kuukausittaiset raportit saa tehtyä näistä ilmaisista sovelluksista. Näin suurin osa asiakkaista voi ohittaa ison osan API-dokumentaation toiminnoista. Lisäksi osa API-dokumentaatiota on pankkikortin tallentamiseen liittyviä asioita, joille ei ole kaikille käyttöä. Uskon, että pankkikortin tallentaminen jokaiseen verkkokauppaan erikseen, on vanheneva palvelu, jota tarvitaan vain kuukausittain toistuvissa maksuissa. Tulevaisuuden vaihtoehtona on Google Pay ja Apple Pay tyyppiset palvelut, jossa maksukortti tallennetaan vain yhteen paikkaan, mutta voidaan käyttää kaikissa palveluissa, missä se on maksutapana tarjolla. Dokumentaation lisäksi kannattaa panostaa demoihin ja avoimiin kirjastoihin. Asiakkaalle tulisi tarjota helppo lähtökohta API-rajapinnan liittämiseen. Tässä työssä esitelty Bambora toimii hyvänä esimerkkinä muille. Mielestäni tämän voisi viedä vielä askeleen pidemmälle, demonstroimalla koko maksuprosessi sivustolle upotetulla sovelluksella. Siinä kerrottaisiin mitä API-toimintoja suoritetaan, ja näytettäisiin lähetettävät ja vastaanotettavat tiedot. Eikä olisi mahdollonta, että kyseisen sovelluksen lähdekoodit olisivat myös vapaasti käytettävissä. Yksi asia, joka ei tule esille maksupalveluiden dokumentaatiosta, on kuinka koodit kannattaa jakaa tietoturvallisuuden vuoksi, front- ja backend koodeihin. Siihen löytyy esimerkkejä tämän työn kaavioista.

LÄHTEET

Aaltonen, L. 2019. Pelkkä avainlukulista ei pian enää riitä – Mobiiliavain kannattaa aktivoida viimeistään nyt. WWW-dokumentti. Saatavissa: <https://op.mediatalous/raha-ja-arki/pelkka-avainlukulista-ei-pian-ena-riita-mobiiliavain-kannattaa-aktivoida-viimeistaan-nyt-ce182027022e41e090fe531c2caaafb7> [viitattu 26.4.2020]

Bambora. 2019a. Bambora PayForm (EU) Node library and example page. GitHubin www-sivuilla. Saatavissa: <https://github.com/bambora/payform-node-lib> [viitattu 26.4.2020]

Bambora. 2019b. Node library reference. WWW-dokumentti. Saatavissa: https://payform.bambora.com/docs/web_payments/?page=reference-node [viitattu 26.4.2020]

Checkout Finland. 2019. Checkout PSP API. WWW-dokumentti. Saatavissa: <https://checkoutfinland.github.io/psp-api/#/> [viitattu 26.4.2020]

Checkout Finland. 2020. Maksuratkaisut suoraan mobiilisovellukseen. WWW-dokumentti. Saatavissa: <https://www.checkout.fi/mobiilisovellus> [viitattu 26.4.2020]

Huttunen, K. 2019. Verkkokaupan maksutavat. WWW-dokumentti. Saatavissa: <https://www.zoner.fi/verkkokaupan-maksutavat/> [viitattu 14.2.2020]

Javascript.info. 2020. Promise. WWW-dokumentti. Saatavissa: <https://javascript.info/promise-basics> [viitattu 26.4.2020]

Kivisaari, T. 2016. API:t ovat modernin integraatiostrategian ydin. WWW-dokumentti. Saatavissa: <https://blog.digia.com/rest-api> [viitattu 7.3.2020]

Lintinen P. 2020. ePayment Sales Manager. Haastattelu 3.3.2020. Paytrail. Microsoft. 2010. Data Confidentiality. WWW-dokumentti. Saatavissa: <http://msdn.microsoft.com/enus/library/ff650720.aspx> [viitattu 26.4.2020]

myynti@checkout.fi. 2020. Sähköpostiviesti 12.3.2020. Checkout Finland.

Nuurtamo. E. 2019. Client for checkout payment service. GitHubin www-sivuilla. Saatavissa: <https://github.com/Hobiver/checkout-finland> [viitattu 26.4.2020]

Paytrail Example 4.12. Confirming payment receipt with PHP class. WWW-dokumentti. Saatavissa: <https://docs.paytrail.com/en/ch04s05.html> [viitattu 26.4.2020]

Paytrail. 2014. Paytrail_Module_Rest.php version 2.0. PHP-kirjasto luokka. Saatavissa: https://docs.paytrail.com/files/Paytrail_Module_Rest.zip [viitattu 26.4.2020]

Paytrail. 2019. Integration guide. v5.7. WWW-dokumentti. Saatavissa: <https://docs.paytrail.com/en/index-all.html> [viitattu 26.4.2020]

Paytrail. 2020. Medialle. WWW-dokumentti. Saatavissa: <https://www.paytrail.com/medialle> [viitattu 26.4.2020]

Tainio H. 2020. Sales Manager. Haastattelu 6.3.2020. Bambora.

Vakula M. 2017. Paytrail-master. Saatavissa: <https://github.com/mvakula/paytrail> [viitattu 26.4.2020]

Verkkokauppa Pohjoismaissa 2019. PDF-dokumentti. Saatavissa: <https://www.postnord.fi/siteassets/raportit/verkkokauppa-pohjoismaissa/verkkokauppa-pohjoismaissa-2019.pdf> [viitattu 26.4.2020]

Verkkokauppa Suomessa 2019. PDF-dokumentti. Saatavissa: <https://www.paytrail.com/hubfs/Verkkokauppa-Suomessa-2019.pdf?hsLang=fi> [viitattu 26.4.2020]