

Mike Baumans CCS02

Arne van Beethoven CCS01

Frederik Crauwels CCS03

Jens Mertens CCS04

Seth Peeters CCS04

Yari van Doninck CCS01



Project Hosting: CCS7

PLAN VAN AANPAK

Thomas More Geel, IT Factory Cloud & Cybersecurity, 2020-2021



INHOUDSOPGAVE

| | |
|--|----|
| INHOUDSOPGAVE | 1 |
| INLEIDING | 2 |
| OPDRACHT | 2 |
| Beschrijf de aanleiding of achtergrond van je onderzoek. | 2 |
| De huidige stakeholders en partners zijn: | 3 |
| Verwachte resultaat: | 4 |
| De business case: | 6 |
| PROJECTPLANNING | 7 |
| OS | 7 |
| Security | 8 |
| Multi-Layer Infrastructure | 9 |
| Uptime garanderen | 9 |
| PHP Hosting | 10 |
| Domains opstellen (DNS) | 10 |
| Login-page end user | 10 |
| Platform user-friendly | 11 |
| Back-ups | 11 |
| Versiebeheer | 11 |
| Ticket systeem | 11 |
| Database | 11 |
| INFORMATIEVERZAMELING EN RAPPORTERING | 12 |
| PROJECTTEAM | 13 |
| Welke rollen nemen zij op? | 13 |
| Welke competenties hebben de teamleden? | 13 |
| Wie is verantwoordelijk voor welke onderdelen? | 14 |
| CONCLUSIE | 15 |
| BIJLAGEN | 16 |
| BRONNEN | 19 |

INLEIDING

Terwijl de wereld meer en meer geconnecteerd wordt met het internet moeten de mensen ook volgen. Er is natuurlijk de Cloud waar al veel data op staat, maar is dit alles. Meer en meer bedrijven maken en hosten een website. Hierdoor zijn ze overal en altijd beschikbaar. Er bestaan al bedrijven die hosting service aanbieden maar de vraag is, is het veilig en is het betaalbaar. Natuurlijk kan u zelf ook alles zelf proberen te doen maar dat moet niet meer. Wij CCS7 in samenwerking met Hosting X brengen de nieuwe oplossing een veilig en betaalbaar hosting platform voor iedereen.

Wij willen zorgen dat iedereen een veilig en betaalbaar platform heeft waar zij al hun hosting op kunnen doen. Natuurlijk willen wij hierdoor de klant dichterbij het bedrijf brengen maar ook ontwikkelaars te helpen om hun testen online te kunnen doen om zo tot een betere wereld te komen.

In dit document gaan we alles al toelichten, hier zit een structuur in. Eerst gaan we de opdracht van Hosting X toelichten. Hierin komt u meer te weten over ons onderzoek, het resultaat dat wij verwachten en onze business case. Hierna wordt er meer informatie gegeven over projectplan, dit omschrijft hoe wij ons project gaan aanpakken, welke taken ons project heeft. Vervolgens geven we meer informatie over hoe en wanneer wij contacteren met de betrokken partijen. Daarna komt ons projectteam hierin vind u onze rollen structuur en onze competenties terug. Natuurlijk hebben we ook nog een conclusie waarin we bespreken wat we geleerd en gerealiseerd hebben. Als laatste hebben we ook nog onze bijlage waarheen er in document verwijst wordt en onze bronnenopgave waarin al onze bronnen van ons onderzoek komen te staan.

OPDRACHT

Beschrijf de aanleiding of achtergrond van je onderzoek.

We voeren het project uit voor **studenten van 2 ITF en bedrijf hosting X**. Het project houdt in dat we een hosting platform aanbieden die PHP / MySQL applicaties ondersteunt, alsook een (S)FTP methode aanbieden.

Elk team voert het project zo goed mogelijk uit. De opdracht omvat een enorme leermogelijkheid voor alle teamleden, als ook een ideale oplossing voor de klant. Het hosting project draait vooral om een praktische hands-on aanpak waarbij er 3 delen zijn om uit te voeren.

- **Deel 1: studiewerk (conceptfase).**

Welke functionaliteiten moet het project bevatten? Een inleidende presentatie door intraceto, onderzoek naar technologieën en overtuigen van het management a.h.v. een presentatie.

- **Deel 2: implementatie gedeelte.**

De gekozen technologie op echte apparatuur op school laten werken om een eindresultaat aan de klant te tonen. Minstens 12 van de 32 niet-functionele eisen die beschreven staan in Ops Report Card moeten aanwezig zijn. Aan het einde is er evaluatie.

- **Deel 3: testfase.**

Tijdens de projectweek zal het team bestanden aangeleverd krijgen van klanten. Deze moeten om 24 uur tijd geïntegreerd worden in het project.

De opdrachtgever ervaart het probleem dat er geen juiste oplossing is voor het huidige probleem ; namelijk hosting. De huidige oplossingen zijn: niet gratis, niet goedkoop of niet beschikbaar voor deze doeleinden binnen de opleiding. Voor de IT-studenten is er ook een kans om iets bij te leren als ook een resultaat te voorzien voor het gegeven probleem, als ook toekomstige problemen. Zo kan dit project zowel een oplossing voor nu zijn als ook voor de toekomst en toekomstige eindgebruikers (2ITF studenten).

De huidige stakeholders en partners zijn:

- **Intracto (als externe ondersteuning).**
- **Hosting X: Het bedrijf dat de wedstrijd doet voor een hosting platform.**
- **Leerkrachten van Thomas More (opdrachtgevers, als ook mogelijke gebruikers).**
 - Filippo Bagnoli.
 - Bram Verbruggen.
 - Filip Schouwenaars.
 - Nathalie Bosschaerts.
 - Gunther Van Landeghem.
- **Studenten van 2 ITF**
 - CCS / IoT studenten die aan het project deelnemen om het probleem op te lossen.
 - Alle andere studenten die als eindgebruikers kunnen gezien worden van het hosting platform.
- **Het project team: in dit geval CCS7:**
 - Frederik Crauwels.
 - Jens Mertens.
 - Mike Baumans.
 - Seth Peeters.
 - Yari Van Doninck.
 - Arne Van Beethoven.

Hierbij zijn de voornaamste stakeholders en partners vernoemd, wat niet uitsluit dat er nog andere toekomstige eindgebruikers kunnen bijkomen, bijvoorbeeld.

Op het vlak van datacenters, security, Ops Report Card en hosting staan we nog maar aan het begin van onze capaciteiten. Zowel het vak PHP als SQL hebben we gehad, waardoor we al een idee hebben van wat het hosting platform moet kunnen aanbieden. FTP methodes hebben we ook al meerdere keren moeten gebruiken maar hier begint dan ook alles.

We hebben voor dit project altijd andere software / hardware mogen gebruiken om iets te hosten, uploaden,... en nu gaan we dit volledig zelf mogen uitwerken en hosten, uploaden,...

We staan op het punt om een plan van aanpak te maken als ook verder onderzoek uit te voeren naar technologieën om te gebruiken voor het project. Daarnaast wordt er ook nog verdere ondersteuning geboden door Intracto, de leerkrachten en verdere opleiding doorheen verschillende opleidingsonderdelen.

Verwachte resultaat:

Aan welke 'eisen' moet de oplossing voldoen?

Een eerste eis is dat ons webplatform goed beveiligd moet zijn. Wanneer een klant zijn php project online wil zetten bij ons, moet dit natuurlijk goed beveiligd zijn, zodat alleen de klant eraan kan.

Vervolgens willen we ons platform zo gebruiksvriendelijk mogelijk maken. Niet elke klant heeft evenveel ervaring met web hosting. Daarom willen wij een platform aanbieden dat zeer makkelijk te gebruiken is zodat onze klanten meteen aan de slag kunnen. Indien ons platform niet duidelijk en te complex is, kan dit een afknapper zijn voor een potentiële klant.

Waar moet rekening mee gehouden worden?

Uiteraard moeten er ook met een aantal zaken rekening gehouden worden. Zoals bijvoorbeeld met de availability van onze service. We moeten ervoor zorgen dat we zo'n optimaal mogelijk uptime garanderen zodat de klanten niet in de problemen komen te zitten. Een te lange downtime kan zeer problematisch zijn en dit willen we ten allen tijden vermijden.

Ook moeten we rekening houden met het maken van verschillende backups. Indien er zich een probleem voordoet dat niet opgelost geraakt, kunnen we terugschakelen naar een vorige versie dankzij backups die we gemaakt hebben. Door backups te maken creëren we een vangnet voor onszelf waardoor we minder snel of zelfs zo goed als niet in de problemen geraken.

Enkele andere factoren waar rekening mee gehouden moet worden is de gebruiksvriendelijkheid van het platform, de algemene beveiliging van het platform, en het werken met versiebeheer, zodat we meerdere versies kunnen gebruiken.

Hoe moet het eindproduct / de eindervaring eruit zien?

Ons hosting platform moet vooral klantvriendelijk zijn zodat iedere klant/developer met enkele kliks hun PHP project online kan zetten. Zodanig dat de eindgebruiker zonder veel kennis van het hosting gebeuren hun website kan hosten. Er moet zoveel mogelijk met een grafische interface gewerkt worden zodanig dat de klant de achterliggende processen niet hoeft te zien.

Ook zullen we indien we hier extra tijd voor hebben extra services aanbieden. Hierbij moet u denken aan mail services en advertentie services.

Formuleer de doelstelling van je project. Wat wil je met je project bereiken?

Voor ons staat veiligheid en gebruiksvriendelijkheid voorop. We willen er voor zorgen dat onze klanten op een makkelijke manier hun PHP project kunnen hosten. Hiervoor hoeft de klant geen ervaring te hebben in het hosting gebeuren. We zorgen ook voor een FAQ pagina waar veel voorkomende vragen beantwoord zijn. Zodat ook zonder onze directe tussenkomst het makkelijk te gebruiken is voor de gebruiker. Maar voor specifieke vragen is het zeker mogelijk om met ons contact op te nemen. Hierbij zal u direct worden doorverwezen naar de persoon die de juiste kennis heeft om uw probleem op te lossen.

Ook is de beveiliging van de gebruikers belangrijk zodat niemand aan elkaars project kan. In een ideale situatie zou ons hosting platform 24/24 actief zijn en geen downtime hebben. We zullen er voor zorgen dat iedereen dus de juiste rechten krijgt. De gebruiker zal zelf veel kunnen instellen en aanpassen. Maar voor aanpassingen met grote gevolgen zal het nodig zijn om een administrator aan te spreken en in overleg de nodige aanpassingen te doen. Uiteindelijk zou zowel de administrator als de eindgebruiker zo weinig mogelijk input moeten geven om een platform op te starten. We willen dit verzorgen met de automatisatie van enkele onderdelen.

De business case:

Web Hosting wordt steeds meer een cruciaal element in de digitale wereld van vandaag. E-commerce en Social Media platforms worden alsmaar populairder en zorgen steeds meer voor een dynamische en gebruiksvriendelijke Internet ervaring. Maar door de exponentiële groei van deze digitale wereld en het blijvende groeiende aanbod aan internetplatforms is de nood aan security van al de data, van zowel klanten/gebruikers als beheerders van deze internetplatforms, van cruciaal belang.

Elk jaar worden er meer en meer security threats vastgesteld. Deze threads worden ook elk jaar geavanceerder en moeilijker om zowel te ontdekken, als tegen te houden. Maar het blijft enorm belangrijk dat de data op het internet zijn integriteit behoudt en niet voor malafide doeleinden wordt gebruikt.

Een Web Hosting platform dat beheerd wordt door IT-professionals is hier de ideale oplossing voor. In plaats van dat een organisatie zelf zijn digitale data probeert de beveiligen zorgt een goed web hosting platform ervoor dat dit automatisch voor de klant gebeurt. Dit zorgt voor een veel kleinere kans op data inbreuken bij een organisatie. Een web hosting platform is speciaal ontwikkeld om een veilige en gebruiksvriendelijke omgeving te creëren voor organisaties. De organisaties moeten dan niet zelf met de onnodige complicaties omgaan die bij de wereld van cybersecurity en het hosten van hun internetplatform horen. Het kost ook heel veel voor een organisatie om dit volledig zelf te doen en als er dan toch iets misloopt worden de kosten alsmaar groter. Daarom maken heel veel organisaties gebruik van deze web hosting platforms die beheerd worden door IT-professionals. Zo kunnen ze voor een betaalbare prijs op twee oren slapen 's nachts.

Het is dus echt wel economisch interessant om een goed functionerend en betrouwbaar web hosting platform op te bouwen. De vraag ernaar stijgt elk jaar exponentieel met de groei van de digitale wereld, en in deze digitale wereld is het voor organisaties erg belangrijk dat hun data veilig gewaarborgd is, maar ook toegankelijk is voor de juiste mensen. Web Hosting vormt de ideale oplossing hiervoor.

PROJECTPLANNING

Wij gaan ons project opdelen in meerdere delen, hiermee kunnen we agile plannen en werken. Een van onze grootste taken is het onderzoek waarmee we beginnen in de conceptfase. Hierin gaan we opzoek naar verschillende programma's, software en technieken om ons hosting platform te kunnen maken. Na het onderzoek gaan we naar de testfase, waarin we alles goed testen en kijken of het samen werkt met de opties voor andere delen. Na het testen gaan we documenteren, hierin komen de voor- en nadelen over elk programma, software en techniek zodat wij als team kunnen beslissen wat we gaan gebruiken voor onze volgende stap. Als laatste gaan we alle verschillende programma's, software en technieken toepassen en op punt brengen. Zodat we een mooi eindproduct aan onze klanten kunnen afleveren. Op het einde gaan we implementeren, hierin gaan we alle programma's, software en technieken toepassen, op punt brengen en klaar maken voor onze klanten.

Voor onze projectplanning op te stellen hebben we gebruik gemaakt van een WRM-schema en een planning in Excel. Het WRM-schema (Tabel 2), de planning in Excel (Tabel 1) en de tijdsschatting per milestone (Tabel 3) kan je vinden in onze bijlage.

OS

Voor onze OS gaan wij duidelijk kiezen voor een unix / linux besturingssysteem (bijlage figuur 1). Bij voorkeur verkiezen wij hieruit een linux besturingssysteem met het pakket van ubuntu (bijlage figuur 2). De server variant wordt gebruikt i.p.v. de desktop variant.

De grootste voordelen:

- **Open-source:** het is gratis en kan op allerlei manieren zelf aangepast worden.
 - **Gratis:** het is géén betalend besturingssysteem (windows wel).
 - **Aanpassingen maken:** met een open-source besturingssysteem kan je zelf heel veel aanpassingen maken op vlak van security, bijvoorbeeld. Je hebt hier veel vrijheid in als developer.
- **Gebruiksvriendelijk:** De meerderheid van de webserver draait al op unix , linux met een ubuntu pakket als ondersteuning. Hierdoor zijn de meeste eindgebruikers en developers bekend met dit systeem.
- **Ondersteuning:** veel programmeertalen worden ondersteund in linux. PHP staat hier zeker en vast bij. Daarnaast gebruikt linux de MySQL database. Je hebt ook een tal van mogelijkheden om een visuele control panel te gebruiken voor hosting.

Security

De keuze voor onze OS wordt duidelijk een linux pakket. De voordelen zijn hier voornamelijk dat het **open-source** is als ook **gratis** en **zeer aanpasbaar**. De keerzijde is echter ook dat onze software open-source is... De code is openlijk beschikbaar voor hackers om deze in te kijken om zo een besturingssysteem of stukje software binnen te raken. Alles is beschikbaar voor iedereen! Zowel voor ons als voor potentiële hackers. De verantwoordelijkheid voor beveiliging ligt dan ook volledig in de handen van de ondernemers, wij dus. Wij zullen maximaal op security moeten inzetten zodat onze eindgebruikers veilig het platform kunnen gebruiken.

We hebben de keuze uit enkele opties om onze beveiliging te maximaliseren:

- **Bekende gevaren:** In de eerste stap zoeken we altijd naar “bekend terrein”. We gaan op zoek naar bekende gevallen van inbraken en hacking. exploit-db, vulldb,... zijn allemaal databases met bekende vulnerabilities voor een stukje software. Dit zijn de meest bekende achterpoortjes om te voorkomen dat anderen het platform kunnen hacken.
- **Consistent updaten:** Een tweede belangrijk aspect is het consistent updaten van de software. De nieuwe security patches moeten voorzien worden voor onze software zodat oude vulnerabilities zo veel mogelijk verwijderd worden. Deze updates zullen voornamelijk security updates zijn. Hierbij zouden we er ook voor moeten zorgen dat niet alle gebruikers zomaar alles kunnen downloaden - enkel een “sudo” user of admin.
- **Linux hardening:** door de aanpasbaarheid van open-source software kunnen we maximaal inzetten op de beveiliging van ons linux pakket. SELinux of AppArmor zijn speciale “security modes” voor ons linux pakket. Daarnaast kunnen we ook additionele opties voorzien zoals:
 - Geen “lege wachtwoorden” voor een account.
 - Geen root login.
 - Poorten dicht maken (DvD, USB, CDrom,...) die niet gebruikt worden. - externe media beschermen.
 - Enkel security patches toestaan (en geen andere installs).
 - Dedicated Apache / Nginx / PHP server bescherming.
- **Backups:** Ook backups zijn een belangrijk aspect bij security. Indien er iets mis gaat moet er altijd een backup gemaakt kunnen worden. Verwacht altijd het onverwachte!
- **Linux Firewall:** Hiermee beschermen we ons platform en de eindgebruikers van de “grote boze WAN”. PfSense is een goede eerste stap - dit is ook open-source en volledig beschikbaar. We willen iedereen beschermen - onszelf (het platform) als het platform van onze eindgebruikers.

Multi-Layer Infrastructure

Multi-Layer Infrastructure is een manier om zowel de infrastructuur als al de software die nodig is om een bepaalde applicatie, server, hosting platform... op te bouwen of te onderhouden op te delen in meerdere stukken, layers meerbepaald. Dus de architectuur van de volledige software van een IT-gerelateerd project bestaat niet uit één grote blok software maar uit verschillende kleinere stukjes software die verbonden zijn met elkaar en zo succesvol en efficiënt het geheel van de IT-applicatie doen werken.

Hoe ziet dit er toegepast op onze situatie uit?

Deze functionaliteit is zowel interessant voor de klant als voor de hosts zelf. Wanneer het systeem mooi opgedeeld is in verschillende samenwerkende componenten is het vele gebruiksvriendelijker om een component dat outdated is of moet vervangen worden weg te halen uit de volledige architectuur zonder dat al de andere componenten hinder ondervinden.

We kunnen zo ook een real-time omgeving creëren en een development omgeving waarin een kopie staat van de real-time omgeving waarin de volledige architectuur in opgebouwd is. Wanneer er dan iets mis gaat in de real-time omgeving kunnen we het component waar er een fout is in de development omgeving gaan debuggen zonder dat we de real-time omgeving moeten deactiveren. Wanneer er een werkende oplossing is in de development omgeving kunnen we deze dan doorvoeren naar de real-time omgeving zonder dat deze hinder heeft ervaren.

Uptime garanderen

Het totale percentage van uptime per maand moet zo hoog mogelijk zijn. Dat wil zeggen dat de sites van klanten liefst bijna altijd beschikbaar zijn. De klant en zijn gebruikers willen graag dat hun site in alle tijden bereikbaar is ook al moeten er aanpassingen gebeuren aan de onderliggende architectuur. Geplande downtime is acceptabel als dit goed gecommuniceerd wordt door de hosts, maar deze zou zo kort mogelijk moeten zijn zodat er niet lang hinder ondervonden wordt.

Hoe ziet dit er toegepast op onze situatie uit?

Wanneer de belangrijkste onderliggende componenten van onze hele architectuur van het hosting platform goed opgebouwd zijn en goed functioneren zal onze uptime in percentage hoog liggen. Als er voldoende aandacht is besteed aan security, implementeren van back-ups, en andere componenten die ons platform moeten beschermen tegen mogelijke doemscenario's zullen er weinig downtime momenten zijn. Zowel de klant als wij, de hosts, kunnen dan met een gerust hart slapen.

PHP Hosting

Een belangrijk onderdeel van een hosting platform is natuurlijk dat we de files van de klanten online kunnen krijgen. In dit geval zijn dat PHP files. Onze klanten beschikken over PHP files, zij willen deze online krijgen dus sturen ze deze door naar ons.

Hoe ziet dit er toegepast op onze situatie uit?

We beginnen allereerst met het opstarten van een OS, hiervoor gaan wij linux gebruiken. Vervolgens als het OS gekozen is, moeten we gebruik gaan maken van een webserver. Hiervoor is er keuze uit verschillende webserver zoals bijvoorbeeld apache, nginx, lighttpd, etc. Na het kiezen/installeren van de webserver moeten we PHP nog installeren zodat dit herkent wordt bij de bestanden.

Als de installaties voltooid zijn kunnen we beginnen aan de configuratie. Dit houdt in dat het IP van de webserver in orde wordt gebracht en de files op de juiste locatie staan etc. Als alles goed geconfigureerd is staat het PHP project normaal online.

Domains opstellen (DNS)

Wanneer alles klaar staat (webserver etc.) kunnen we een DNS server gaan toevoegen. De bedoeling hiervan is dat de klanten hun website niet een nummer als naam heeft (bv. 192.168.1.8) maar een degelijke naam (bv. www.gekozennaam.be). De DNS server gaat dus als het ware dit nummer vertalen naar een ander adres.

Hoe ziet dit er toegepast op onze situatie uit?

Als eerste moeten we de keuze maken uit welke DNS server we gaan gebruiken: bijvoorbeeld cloudflare, google, openDNS, etc. Na het kiezen van de DNS server kunnen we gaan configureren en dit opstellen in de config files. Na een succesvolle configuratie is het IP-adres verandert naar een ander adres.

Login-page end user

De login-page wordt gebruikt zodat de end users (klanten) makkelijk toegang hebben op verdere functies op het platform. Hierbij zouden ze simpel een nieuwe site kunnen maken of een overzicht zien van al hun sites.

Dit gaan wij toepassen met een combinatie van HTML, CSS en PHP. Zodat de inlog/registratie ook user-friendly is en alle users in een database komen.

Platform user-friendly

Een user-friendly platform is zeer belangrijk. Wij willen dat iedereen een goede ervaring heeft bij het gebruik van onze service zodat ze deze gaan blijven gebruiken en aan anderen aanraden.

Wij gaan dit realiseren door een goede kleurencombinatie (bijlage Figuur 4) te kiezen waardoor alles goed leesbaar blijft. Natuurlijk wordt er ook gekeken naar het lettertype en lettergrootte. We zorgen er ook voor dat de site simpel blijft zodat iedereen makkelijk aan hosting kan doen bij ons.

Back-ups

Om ervoor te zorgen dat klanten nooit hun volledige project kwijt zijn zullen we voor back-ups zorgen. Om dit te verzorgen gaan we gebruik maken van GitHub. Dit is volgens ons een goed platform om de back-ups op te doen. Zo kunnen klanten ook zelf aan hun eigen back-ups moest dit nodig zijn.

We zullen een automatisch systeem opzetten dat de gehoste projecten elke x aantal dagen geback-up worden. En er altijd twee vorige versies beschikbaar zijn op de git repo van de klant.

Versiebeheer

We willen dat al onze klanten hun project kunnen hosten onafhankelijk van de versie die ze gebruiken. Hiervoor zullen we opnieuw GitHub gebruiken. GitHub is ook handig voor het monitoren van wat er gebeurt met elk project. we kunnen zien wie wat waar en wanneer aanpast door middel van de monitoring van GitHub zelf. Zo zal voor ons Git veel problemen met versiebeheer oplossen.

Ticket systeem

Als gekozen ticketsysteem gebruiken we UvDesk. Wij kiezen voor deze service omdat deze grotendeels open source is, eenvoudig te customizen is en gebouwd is om eenvoudig bruikbaar te zijn met eenzelfde type LAMP stack als die wij gebruiken.

Database

We onderhouden verschillende databases waar alle data die ons platform nodig heeft wordt opgeslagen, zowel klantendata, performance data en platform informatie. We gebruiken hiervoor MySQL vanwege de compatibiliteit met de LAMP stack en onze eigen ervaringen met SQL.

INFORMATIEVERZAMELING EN RAPPORTERING

Om zeker te zijn dat ons project voldoende bruikbaarheid en klantvriendelijkheid behoudt is het vergaren van feedback en het verzamelen van informatie een belangrijke doelstelling.

Ook is het belangrijk dat de klant vlot geholpen wordt bij het tegenkomen van problemen of vragen met de hosting service.

Hiervoor gaan wij verschillende systemen inbouwen waardoor onze gebruikers hun ervaringen en problemen kunnen delen om zo onze hosting service verder te verbeteren.

Dit doen wij door middel van een ticket systeem, waarbij de gebruikers een support formulier kunnen invullen met hun feedback, dat wordt ingevoerd in het numerieke systeem en gekoppeld blijft aan het account van de gebruiker om het systeem overzichtelijk te houden.

Ideaal worden ook vragen die binnenkomen via alternatieve communicatiekanalen zoals email of social media ook doorgestroomd naar het ticketsysteem.

Hoe wordt er gerapporteerd aan de belanghebbenden?

Met andere stakeholders zoals de leerkrachten en het CCS7 team zelf kan gecommuniceerd worden via de daarvoor voorziene teams kanalen. Met onze gekozen gebruikersgroep (de leerlingen van project PHP) wordt via discord en teams contact gehouden.

Elke week wordt de trello planning (waarin alle begonnen en behaalde taken worden opgelijst) geupload naar de project hosting canvas.

PROJECTTEAM

Welke rollen nemen wij op?

- Documentatieverantwoordelijke: verstuurt de documentatie, zorgt dat deze samenzitten en blijven. Documentatie is volledig en compleet. Planning doorsturen elke week. **Yari Van Doninck & Jens Mertens.**
- Spellingchecker: iedereen moet dit zeker wel doen. Kijk werk na van andere groepsleden in feedback - controle. **De hele groep - dubbelcheck in word en nalezen!**
- Teamleader: Orde in het team. Beslissingen worden doorgehakt,, problemen oplossen binnen het team. Team check, overlopen met bord / planning. Begin (standup meeting) - afsluit (samenvatting). **Frederik Crauwels.**
- Timekeeper: deadlines in het oog houden, agenda in het oog houden en iedereen er op wijzen. Kijk bij de teamleden of deadline lukt. **Arne Van Beethoven & Seth Peeters.**
- Contactpersoon bedrijf / leerkrachten / opdrachtgever: beslist in de groep - maar is de centrale contactpersoon met "de buitenwereld". **Mike Baumans.**

Welke competenties hebben de teamleden?

Iedereen heeft dezelfde competenties omdat we in dezelfde opleiding zitten. We hebben de puntjes verdeeld naargelang de interesse voor het onderdeel. Iedereen heeft zo zijn eigen stukje gekregen om aan te werken tijdens het project.

Op die manier willen we ervoor zorgen dat iedereen specifieke competenties behaald op een deelthema. Vervolgens is het ook de bedoeling dat iedereen samen aan het gehele project werkt. Indien iets niet lukt moet het team beschikbaar zijn om samen aan een deelthema te werken.

Zo willen we er optimaal voor zorgen dat iedereen een specifieke kennis behaald op een deelthema als ook samen extra kennis opdoet over andere thema's. Zo leren we samen verschillende competenties bij op het vlak van IT.

Naast IT competenties leren we ook soft skills aan. Door in team te werken vragen we aan ieder lid om efficiënt in groep te werken als ook verzorgde communicatie te voorzien. Het geven van een powerpoint aan een publiek hoort hier ook bij. Verzorgde documentatie voorzien, samen nakijken en op tijd doorsturen zijn ook belangrijke skills.

Wie is verantwoordelijk voor welke onderdelen?

- **Frederik Crauwels:**
 - Security: monitoring en toepassing.
 - OS.
 - OpsReportCard: security practices (28, 29, 30, 31).
- **Jens Mertens:**
 - Platform voor eindgebruiker maken.
 - Login pagina front-end.
 - Automation.
 - OpsReportCard: Operational practices (11), Fleet management processes (19) en security practices(32).
- **Yari Van Doninck**
 - Multi-layer infrastructure.
 - Uptime garanderen - met monitoring.
 - OpsReportCard: Disaster preparation practices(23), modern team practices (10).
- **Mike Baumans:**
 - Database (+ monitoring).
 - Ticketsysteem (+ monitoring).
 - OpsReportCard: Public Facing Practices (1), modern team practices (5) en automation practices (18).
- **Seth Peeters:**
 - PHP hosting.
 - DNS.
 - Extra hulp.
 - OpsReportCard: Automation practices (16), modern team practices (6).
- **Arne Van Beethoven:**
 - Backups (+ monitoring).
 - Versiebeheer (+ FTP - github).
 - OpsReportCard: Modern team practices (6), Disaster preparation practices (25).

CONCLUSIE

Een hosting platform vereist veel verschillende aspecten om uit te werken. Uit dit vooronderzoek en de opstelling van het plan van aanpak hebben we hier een duidelijker beeld over gekregen.

Een volledig project van A tot Z uitwerken vereist veel (team)werk en inspanning om af te werken. We gaan op deze manier veel nieuwe competenties bijleren: het gebruik van een datacenter, hoe maak je een hosting platform, security, database, PHP, FTP, visualiseren van een login pagina, ticketsysteem,...

Een eerste stap was om dit allemaal te onderzoeken. Na dit onderzoek gaan we dit allemaal in de praktijk omzetten. Met behulp van de leerkrachten en coaches gaan wij zo voor een optimale eindoplossing. Een dergelijk project vereist veel planning en onderzoek om alles goed en juist te laten verlopen. Daarom is dit plan van aanpak een essentieel deel in de uitvoering van ons project.

Zo hopen we samen als team, CCS7, een succesvol project af te ronden voor onze mede-studenten zodat zij zonder problemen van ons hosting platform gebruik kunnen maken.

BIJLAGEN

| 1 | Naam persoon | Eerste revisie | Tweede revisie | Laatste revisie | | |
|----|---|--------------------------------------|--|-----------------|------------------------|---------------------------------------|
| 2 | Plan van aanpak | | 17/03/2021 | 24/03/2021 | 26/03/2021 | |
| 3 | | | | | | |
| 4 | Problemen: | Onderzoeken | Testen | Documenteren | Implementeren | |
| 5 | OS | Frederik Crauwels | | | | |
| 6 | Security | Frederik Crauwels | | | | |
| 7 | Ticket system | Mike Baumanns | | | | |
| 8 | Database | Mike Baumanns | | | | |
| 9 | Backups | Arne Van Beethoven | | | | |
| 10 | Multi-layer infrastructure | Yari Van Doninck | | | | |
| 11 | Uptime garanderen | Yari Van Doninck | | | | |
| 12 | Login page front-end (klant) | Jens Mertens | | | | |
| 13 | Platform user-friendly | Jens Mertens | | | | |
| 14 | Domains opstellen (DNS) | Seth Peeters | | | | |
| 15 | Versiebeheer met verschillende systemen | Arne Van Beethoven | | | | |
| 16 | PHP hosting | Seth Peeters | | | | |
| 17 | | | | | | |
| 18 | | | | | | |
| 19 | | | | | | |
| 20 | | | | | | |
| 21 | Datum | Doelstelling | Tegen | Tijdsbesteding | | |
| 22 | Woensdag 31/3 | Plan van aanpak indienen | 23:59 | | | |
| 23 | | Peer evaluatie | 23:59 | | | |
| 24 | Woensdag 21/4 | | | | | |
| 25 | | | | | | |
| 26 | | | | | | |
| 27 | Teamid | Woensdag 21/4 | Woensdag 28/4 | Woensdag 5/5 | Woensdag 12/5 | Woensdag 19/5 |
| 28 | Frederik Crauwels | Testomgeving opstellen en klaarmaken | Security | Security | Security | Security (monitoring) |
| 29 | Mike Baumanns | Testomgeving opstellen en klaarmaken | Database | Database | Database | Security (monitoring) |
| 30 | Arne Van Beethoven | Testomgeving opstellen en klaarmaken | Backups | Backups | Backups / versiebeheer | Ticketsysteem |
| 31 | Yari Van Doninck | Testomgeving opstellen en klaarmaken | Meerdere omgevingen klaarmaken (multi-layer) | Multi-layer | Multi-layer | Monitoring (Database / ticket system) |
| 32 | Jens Mertens | Testomgeving opstellen en klaarmaken | Platform voor eindgebruiker maken | Platform | Platform | Monitoring (Backups) |
| 33 | Seth Peeters | Testomgeving opstellen en klaarmaken | PHP hosting | PHP hosting | DNS | Monitoring (uptime) |
| | | | | | | Documenteren in orde maken |
| | | | | | | Automation |
| | | | | | | Extra hulp |

Tabel 1: tijdsplan

| Eigenschappen | Wegings-factor |
|--|----------------|
| OS | 5 |
| Security | 3 |
| Ticket system | 2 |
| Database | 4 |
| Backups | 3 |
| Multi-layer infrastructure | 2 |
| Uptime garanderen | 3 |
| Login page front-end (klant) | 1 |
| Platform user-friendly | 3 |
| Domains opstellen (DNS) | 3 |
| Versiebeheer met verschillende systemen | 2 |
| PHP hosting | 4 |
| FTP | 4 |

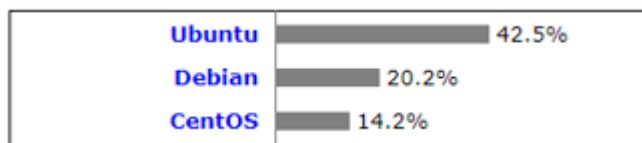
Tabel 2: WRM-schema

| Eigenschappen | Tijdsbestek |
|------------------------------|---|
| OS | 1 ^e keer: 3 uur. Volgende keren: 15 minuten. (met template) |
| Security | Firewall: 3 uur. Linux hardening: 2 dag. Updaten: 1 uur. Bijwerken van alle user stories kan extra tijd nemen. |
| Ticket system | Installatie: 1 uur. Opstellen / design: 2 uur. Customization: 5 uur. Updaten: 30 minuten. |
| Database | Installatie: 1 uur. Opstellen / design: 1 dag. Updaten: 30 minuten. Extra features: Monitoring (wachtwoorden zelf veranderen,...): 2 uur. |
| Backups | Github aanmaken: 1 uur. Linken: 1 uur. Updaten: 30 minuten. |
| Multi-layer infrastructure | Bestaande system "kopieren": 1 uur. Updaten / monitoren: 1 uur. Testing: 2 uur. |
| Uptime garanderen | Combinatie van andere aspecten: security, multi-layer infrastructure, ticket system, backups. (veel tijd.) Installatie / uitwerking: 1 dag. Monitoren: 2 uur. |
| Login page front-end (klant) | Simpele login file / registratie form : 1 uur. Customizen: 12 uur. |
| Platform user-friendly | Design / leesbaarheid : 2 uur. Controleren : 1 uur. Per stukje niet zo lang – het geheel moet gecontroleerd worden wat langer kan duren dan initieel gepland. |
| Domains opstellen (DNS) | DNS kiezen : 1 uur. Configuratie : 2 uur. Aanpassing / wijziging: 1 uur. |
| Versiebeheer | Updaten / veranderen van versie : 2 uur. Verschillende versies beschikbaar stellen: 12 uur. |
| PHP hosting | STACK installatie : 3 uur. Configuratie : 2 uur. Testen : 1 uur. |
| FTP | Via github (?). Configuratie : 1 uur. |

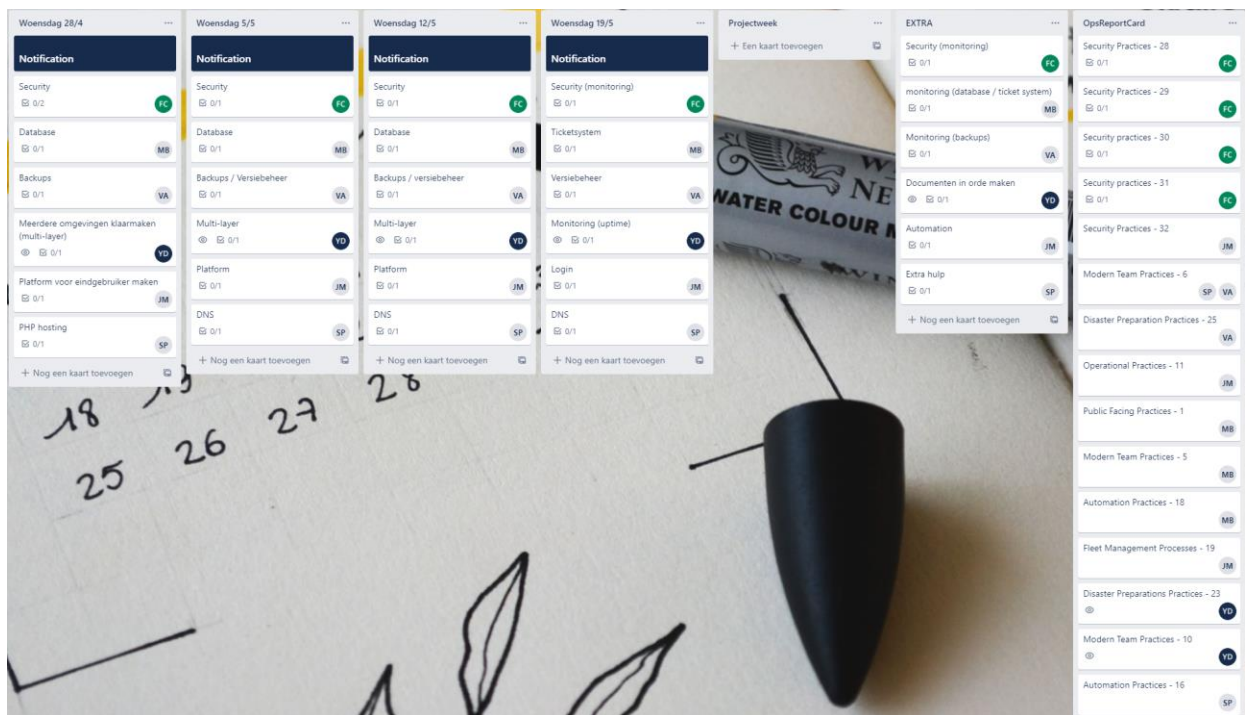
Tabel 3: tijdsschatting per milestone

| | 2020 1 Mar | 2020 1 Apr | 2020 1 May | 2020 1 Jun | 2020 1 Jul | 2020 1 Aug | 2020 1 Sep | 2020 1 Oct | 2020 1 Nov | 2020 1 Dec | 2021 1 Jan | 2021 1 Feb | 2021 1 Mar | 2021 22 Mar |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|----------------|
| Unix | 71.3% | 69.8% | 70.2% | 71.3% | 70.5% | 70.3% | 70.6% | 71.0% | 71.6% | 72.0% | 72.5% | 73.0% | 74.1% | 74.2% |
| Windows | 28.7% | 30.2% | 29.8% | 28.8% | 29.5% | 29.7% | 29.4% | 29.1% | 28.4% | 28.1% | 27.6% | 27.1% | 26.0% | 25.9% |

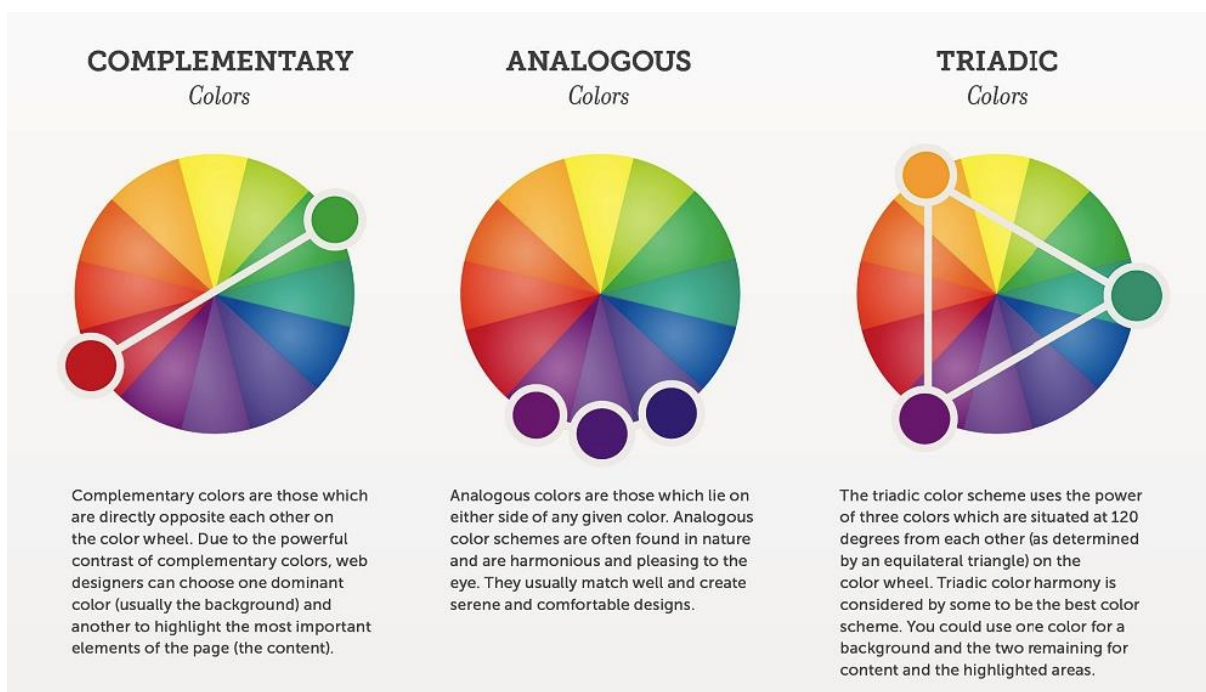
Figuur 1: Unix vs Windows



Figuur 2: Unix Operating Systems



Figuur 3: Projectplanning in trello



Figuur 4: Kleuren contrasten

BRONNEN

Alexandra Leslie. (z.d.). Linux Hosting vs. Windows Hosting (2021): 6 Differences & 6 Best Hosts. hostingadvice.com. Geraadpleegd op 30 maart 2021, van <https://www.hostingadvice.com/how-to/linux-hosting-vs-windows-hosting/>

Gite, V. (2019, 14 augustus). 40 Linux Server Hardening Security Tips [2019 edition]. NixCraft. <https://www.cyberciti.biz/tips/linux-security.html>

Installing SELinux. (z.d.). selinuxproject. Geraadpleegd op 30 maart 2021, van <https://selinuxproject.org/page/Guide/Installation>

Kenlon, S., Both, D., & Kenlon, S. (z.d.). Getting started with Linux firewalls. Opensource.Com. Geraadpleegd op 30 maart 2021, van <https://opensource.com/article/20/2/firewall-cheat-sheet>

Khawaja, G. (2016, 26 november). Linux hardening: A 15-step checklist for a secure Linux server. Computerworld. <https://www.computerworld.com/article/3144985/linux-hardening-a-15-step-checklist-for-a-secure-linux-server.html>

S. (2019, 20 september). How Secure Is Open Source Software? SentinelOne. <https://www.sentinelone.com/blog/open-source-security/>

Usage statistics of operating systems for websites. (z.d.). w3techs.com. Geraadpleegd op 30 maart 2021, van https://w3techs.com/technologies/overview/operating_system

Usage statistics of Unix for websites. (z.d.). www.w3techs.com. Geraadpleegd op 30 maart 2021, van <https://w3techs.com/technologies/details/os-unix>

What is the Difference Between Linux and Windows Hosting? | HostGator Support. (z.d.). hostgator.com. Geraadpleegd op 30 maart 2021, van <https://www.hostgator.com/help/article/what-is-the-difference-between-linux-and-windows-hosting>

Which is More Secure: Windows, Linux, or macOS? (2020, 27 oktober). SentinelOne. <https://www.sentinelone.com/blog/which-is-more-secure-windows-linux-or-macos/>

Wikipedia contributors. (2021, 7 maart). AppArmor. Wikipedia. <https://en.wikipedia.org/wiki/AppArmor>

Offensive Security's Exploit Database Archive. (z.d.). Exploit-Db. Geraadpleegd op 30 maart 2021, van <https://www.exploit-db.com/>

Vulnerability Database. (z.d.). Vuldb. Geraadpleegd op 30 maart 2021, van <https://vuldb.com/>

Packt Editorial Staff. (2018b, mei 21). What is a multi layered software architecture? Packt Hub. <https://hub.packtpub.com/what-is-multi-layered-software-architecture/>