



Utveckling av ett databasövervakningsrapporteringsystem

Sophia Bairou

Lärdomsprov

Informationsteknik

2024

Lärdomsprov

Sophia Bairou

Yrkeshögskolan Arcada: Informationsteknik, 2024.

Uppdragsgivare:

Advania Finland

Sammandrag:

Detta lärdomsprov beskriver planeringen och utvecklingen av ett databasövervakningsrapporteringsystem för Advania Finland och deras kunder. Planeringen består av en kravanalys, där både företagets och kundernas krav och behov samlas in. Till planeringen hörde bland annat intervjuer där service managers intervjuades och även andra personer på företaget, för att kunna få mera exakta krav och information för vad som behövs skapas och förbättras till det nya databasövervakningsrapporteringsystemet.

Databasövervakningsrapporteringsystemet existerade redan hos Advania, men var bristfälligt och krävde uppdatering. Kravet till att skapa ett uppdaterat rapporteringssystem kom från Advania Finlands kunder. Både det gamla och det nya vidareutvecklade databasövervakningsrapporteringsystemet är uppbyggda i Zabbix, och skrivet med Python kod. Det vidareutvecklade systemet är uppdaterat med bl.a. ny Python kod för att nå förväntade krav. Även SQL har använts för att bland annat möjliggöra övervakningen av lyckade eller misslyckade säkerhetskopieringar av databaser, vilket också var ett krav som skulle vara synligt i rapporten. För att skapa den visuella delen av rapporten har Power BI använts. För att sedan få veta ifall rapporten var användbar skickades den till Advania Finlands produktutvecklingsavdelning för testning. Efter att den nya rapporten skickades för testning och feedback, visades positiva resultat. Trots att rapporten ännu inte var helt färdigställd vid testtillfället, visade den ändå potentialen att möta användarnas behov och förbättra övervakningsprocessen för Advania Finland och deras kunder. Den nya rapporten innebar en betydande förbättring jämfört med det tidigare bristfälliga systemet. Den levererade en mer detaljerad och exakt överblick över databasövervakningen, vilket gav kunderna ökad insyn och kontroll över sina system. Dessutom möjliggjorde de flexibla filtrerings- och visualiseringsfunktionerna i Power BI en anpassning av rapporten efter användarnas specifika behov och preferenser.

Nyckelord:

Databasövervakningsrapporteringssystem, Advania Finland, kravanalys, planering, databas, Zabbix, Python, SQL

Degree Thesis

Sophia Bairou

Arcada University of Applied Sciences: Information technology, 2024.

Commissioned by:

Advania Finland

Abstract:

This degree thesis describes the planning and development of a database monitoring reporting system for Advania Finland and their clients. The planning phase involves a requirements analysis, gathering the needs and requirements of both the company and its customers. This includes conducting interviews with service managers and other personnel within the company to obtain more precise requirements and information for what needs to be created and improved in the new database monitoring reporting system. While a similar system already existed at Advania, it was lacking and required updating. The demand to create an updated reporting system came from Advania Finland's customers. Both the old and the newly developed database monitoring reporting systems are built on Zabbix and written in Python code. The enhanced system has been updated with new Python code to meet the expected requirements. SQL has also been used to enable monitoring of successful or unsuccessful database backups, which was also a requirement to be visible in the report. Power BI has been used to create the visual part of the report. To determine the usefulness of the report, it was sent to Advania Finland's product development department for testing. After the new report was sent for testing and feedback, the results were positive. Despite the report not being fully completed at the time of testing, it still showed the potential to meet the needs of users and improve the monitoring process for Advania Finland and their clients. The new report represented a significant improvement over the previous insufficient system. It provided a more detailed and precise overview of database monitoring, giving clients increased insight and control. Additionally, the flexible filtering and visualization features in Power BI allowed for customization of the report according to the users specific needs and preferences.

Keywords: Database monitoring reporting system, Advania Finland, requirements analysis, planning, database, Zabbix, Python, SQL

Innehåll

1	Inledning	6
1.1	Bakgrund	6
1.2	Syfte och målsättningar	8
1.3	Metoder	9
1.4	Avgränsningar	10
1.5	Lärdomsprovets struktur	11
2	Kravanalys	12
2.1	Definition av kravanalys	12
2.2	Kravanalys på Advanias rapporteringssystem	14
3	Rapporteringssystemets realisering	15
3.1	Biblioteksval	15
3.2	Python	17
3.3	SQL	18
3.4	Power BI	19
3.5	Linux	20
3.6	Den tekniska utvecklingen av rapporteringssystemet	22
4	Resultat	24
4.1	Slutliga rapporten	24
5	Slutledning och framtida utveckling	26
	Källor	29
	Bilagor	30

Figurer

Figur 1. Nuvarande läget av rapporteringen	7
Figur 2. Exempel på nuvarande grafisk layout på rapporten	7
Figur 3. Prototyp av vad man vill att ska synas i själva rapporten.	14
Figur 4. Zabbix startsida, "dashboard". Här ser man alla kunders övervakning samt övervakningens status med färger i allvarlighetsgrad.....	16
Figur 5. Egengjord "klon" av slutprodukten.	26

1 Inledning

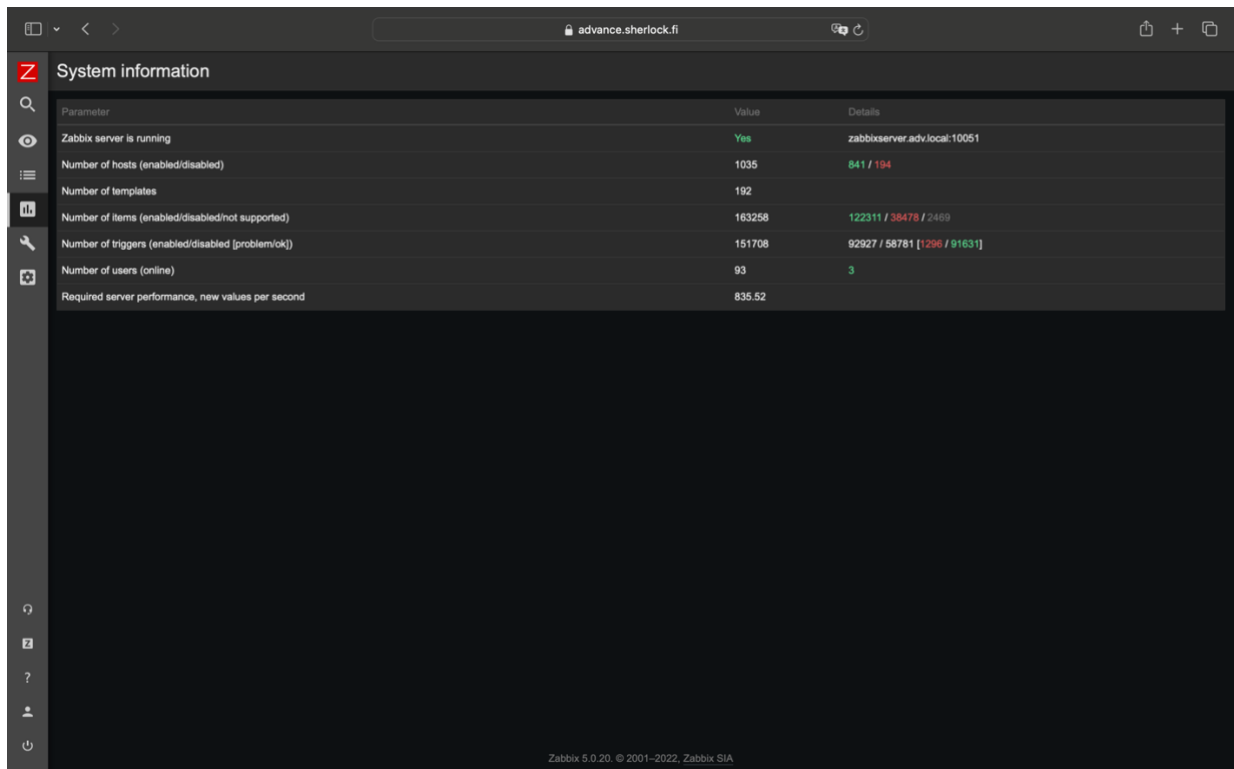
1.1 Bakgrund

Advania är ett ledande IT-företag i Norden som grundades år 1939 i Island (Advania, 2024). Advania har verksamhet i 9 länder, och har allt som allt 42 kontor. I Finland är Advania en viktig del av Advania Group med omkring 350 anställda och 6 kontor runt om i landet. Inom Advania Group är databasavdelningen i Finland en viktig del av IT-infrastrukturverksamheten. Med cirka 100 kunder under sitt ansvar övervakar och underhåller avdelningen en mängd databaser och databasinstanser, som är avgörande för företagets verksamhet.

En utmaning som Advania Finland och dess kunder stött på är det befintliga databasrapporteringsystemet. Det nuvarande systemet som är baserat på Zabbix (Zabbix, 2024), ett open-source program för övervakning, har visat sig vara otillräckligt för kundernas behov. Standardrapporterna från Zabbix saknar den nödvändiga detaljnivån och flexibiliteten för att tillfredsställa Advanias kunders krav och behov. En av orsakerna till detta är att de befintliga rapporterna är hårdkodade, vilket innebär att man saknar möjligheten att anpassa innehållet och utseendet av rapporterna, vilket man tydligt kan se i Figur 1. I dessa rapporter får man inte ut all information man vill ha, utan bara en specifik del av informationen. Man kan inte heller välja att få ut all information i ett och samma dokument, utan allt kommer i separata dokument, presenterat i Figur 2. Detta har skapat problem för servicechefer och kunder eftersom rapporterna inte kan skräddarsys för att passa varje kunds unika behov och önskemål.

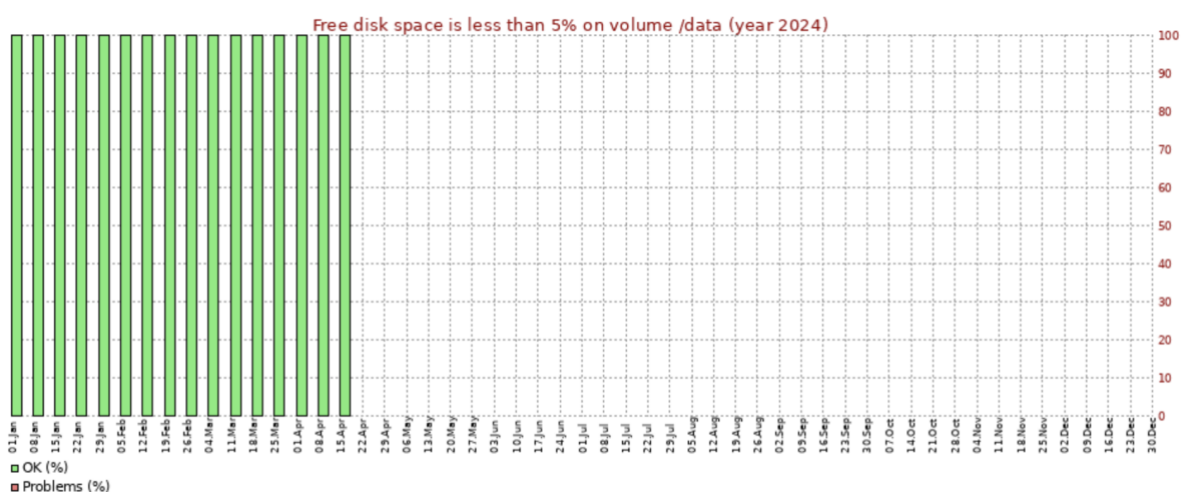
Advania har fungerat som uppdragsgivare för detta lärdomsprov där uppdraget har handlat om att förbättra ett databasrapporteringsystem för uppdragsgivarens kunder. Det gamla systemet har setts som bristfälligt, och kräver en uppgradering. Det gamla systemet saknar en hel del Python kod som krävs för att kunna förbättra databasrapporteringsystemet. Advania har fått önskemål av deras kunder att kunna få en rapport över databasövervakningen och kunderna har alla haft olika önskemål av vad de vill kunna se. Genom att ta itu med denna utmaning på ett framåtriktande sätt visar

Advania sitt engagemang för att leverera högkvalitativa och skräddarsydda lösningar till sina kunder.



Figur 1. Nuvarande läget av rapporteringen

I figur 1 kan man se nuvarande läget av rapporteringen. Rapporteringstyperna är begränsade och uppfyller inte kraven för kunderna.



Figur 2. Exempel på nuvarande grafisk layout på rapporten

I det gamla outvecklade systemet får man inte ut en rapport med all information man vill ha, utan man måste manuellt kombinera rapporterna för att få en bättre insikt. Exempel på en övervakning av en kunds hårddiskens kapacitet.

1.2 Syfte och målsättningar

Syftet med detta lärdomsprov är att utveckla ett system som fungerar som en integrerad service för att underlätta hanteringen av databasrelaterade frågor. Målet är att förbättra kundrelationer och öka försäljningen genom att erbjuda en effektiv och användarvänlig plattform för att övervaka och analysera databaser för varje individuell kund. Genom att skapa ett databasövervakningsrapporteringsystem strävar projektet efter att ge service managers ett verktyg för att snabbt och noggrant utvärdera förändringar i databaser och identifiera potentiella problem eller trender. För att uppnå detta mål kommer systemet att fokusera på att samla in och analysera data om händelser inom databaserna, inklusive användarinteraktioner och programvarurelaterade fel under valda tidsperioder. Genom att noggrant mäta och visualisera dessa trender kommer användarna att kunna fatta välgrundade beslut och vidta åtgärder för att förbättra prestanda och stabilitet i sina databasmiljöer.

Forskningsfrågan som styr detta arbete lyder: " Hur kunde ett rapporteringssystem utvecklas och implementeras för att underlätta processen av att gå igenom databasrelaterade frågor?". Genom att besvara denna fråga kommer projektet att kunna identifiera och adressera de specifika behov och utmaningar som företaget står inför när det gäller hantering av sina databaser.

Det nya systemet representerar en vidareutveckling av det befintliga handskrivna Python-projektet, som redan innehåller handskrivna SQL-frågor. Detta ger projektet en betydande flexibilitet och anpassningsbarhet för att möta Advanias unika krav och behov. Genom att bygga vidare på denna befintliga plattform kommer projektet att kunna dra nytta av den existerande kodbasen samtidigt som det möjliggör för vidareutveckling och förbättringar. Det är viktigt att notera att det nya systemet representerar en unik intellektuell egendom och har potential att övervinna eventuella begränsningar som andra befintliga applikationer kan ha. Genom att utveckla en

skräddarsydd lösning som är specifikt anpassad för företagets behov, strävar projektet efter att skapa ett verktyg som kan maximera effektiviteten och prestandan hos kundernas databaser, samtidigt som det säkerställer att deras immateriella egendom är skyddad.

1.3 Metoder

I arbetet tillämpas flera noggrant utvalda metoder för att säkerställa en grundlig och heltäckande insamling samt analys av data.

Först och främst genomförs en omfattande kravanalys, vilket inkluderar intervjuer med personer inom företaget. Detta innefattar inte bara service managers och databasadministratörer, utan även specialister inom business intelligence, vars insikter och perspektiv är avgörande för att förstå de specifika behoven och kraven för det nya rapporteringssystemet. Dessa intervjuer ger en djupgående förståelse för de olika personernas synpunkter och önskemål, och man tar även hänsyn till direkt feedback från kunderna, vilket är avgörande för att säkerställa att det nya systemet effektivt möter deras behov och förväntningar.

Då data har samlats in genom kravanalysen, tillämpas projektuppföljnings- och sammanfattningsmetoder för att hantera och analysera den insamlade informationen på ett strukturerat sätt. För att effektivt organisera och sammanställa den insamlade informationen används dokumentation samt projektuppföljningsprogrammet monday.com (Monday, 2024), vilket möjliggör en systematisk och överskådlig hantering av relevant data. Genom denna noggranna dataanalysprocess säkerställs att resultatet är tillförlitligt och välgrundat.

Den färdiga rapporten granskas noga av företagets kontaktperson för att säkerställa exakthet och tillförlitlighet. Slutprodukten valideras således av företaget innan den slutgiltigt tas i bruk, vilket ger en extra säkerhetsnivå för att upptäcka och åtgärda

eventuella fel eller brister. Det slutliga godkännandet av rapporten kommer från projektgivar-teamet. De granskar noggrant kundernas önskemål och jämför dem med den slutliga produkten för att säkerställa en perfekt matchning. Efter godkännandet av rapporten kommer produkten att gå igenom en pilot-period där en noggrant utvald grupp av kunder får möjlighet att testa och utvärdera den. Denna pilot-period är avgörande för att säkerställa att produkten fungerar enligt förväntningarna och möter kundernas behov och förväntningar på bästa möjliga sätt. Det är viktigt att notera att denna del av projektet inte kommer att ingå i lärdomsprovet, eftersom den slutgiltiga produkten kommer att tas i bruk efter lärdomsprovet är avslutat. Eftersom forskningen baseras på insamling och analys av företagsdata och inte involverar identifierbara individer eller deras personliga information, så finns det inte några negativa etiska konsekvenser för de berörda parterna. Denna systematiska noggrannhet och uppmärksamhet på etiska överväganden säkerställer att projektet genomförs på ett ansvarsfullt och professionellt sätt.

1.4 Avgränsningar

För att säkerställa att projektet förbättrar övervakningsrapporteringsystemet på ett effektivt och genomtänkt sätt är det nödvändigt att fastställa tydliga gränser och mål. Genom att noggrant definiera Advania Finlands specifika krav och önskemål för förbättring av det nuvarande systemet kan projektet fokusera på att adressera dessa behov utan att förlora fokus eller bli för omfattande. Det är viktigt att understryka att projektet inte strävar efter att bygga ett helt nytt databasövervakningsrapporteringsystem från grunden. Istället är målet att bygga vidare på det befintliga systemet för att förbättra och utöka dess funktionalitet i enlighet med Advania Finlands specifika krav och önskemål.

Genom att göra detta kan projektet dra nytta av den befintliga infrastrukturen och resurserna samtidigt som det möter de utmaningar och behov som identifierats av Advania Finland och deras kunder. Genom att etablera klara och tydliga gränser för projektet kan man undvika att det blir för omfattande. Detta gör det möjligt att fokusera tid och resurser på att lösa de specifika problemen och utmaningarna som Advania Finland står inför när det gäller databasövervakningsrapporteringsystemet.

Genom att eliminera onödig komplexitet kan projektet drivas framåt med en tydlig riktning och målsättning. Denna fokuserade och målinriktade strategi möjliggör en mer detaljerad och målinriktad planering av projektets olika faser och aktiviteter. Genom att arbeta inom tydligt definierade gränser kan man effektivt prioritera uppgifter och resurser för att uppnå de önskade förbättringen och utvecklingen. Slutligen, genom att etablera dessa tydliga gränser och mål för projektet kan man säkerställa en smidig och effektiv arbetsprocess. Detta minskar risken för avvikelser eller förseningar och ökar chanserna till framgångsrika resultat som uppfyller Advania Finlands förväntningar och behov.

Inom Advania görs också stora delar av den s.k. forskningen, dvs. är detta arbete ett specifikt arbete från företaget. Ingen vidare forskning i t.ex. böcker eller andra personers forskningar har behövts. Advania har även fungerat som en källa i detta arbete, dess produktion och utveckling, samt dess interna dokumentation. Allt detta är hemligstämplad pg.a. företagets immaterialrätt.

1.5 Lärdomsprovets struktur

Detta lärdomsprov är strukturerat i tydliga kapitel som reflekterar projektets olika faser och arbetsprocesser för att ge en grundlig och detaljerad insikt i projektets genomförande och resultat.

I kapitel 2 presenteras kravanalysen ingående som en metod för att systematiskt samla in och analysera de krav och den relevanta informationen som är avgörande för projektets framgång. Denna fas är av yttersta vikt för att säkerställa att projektet är välgrundat och att det adresserar de verkliga behoven och önskemålen hos Advania Finland och deras kunder.

Kapitel 3 är dedikerat åt den praktiska realiseringen av själva vidareutvecklingen av databasövervakningsrapporteringssystemet. Här diskuteras använda bibliotek, tekniker och verktyg i detalj för att möta projektets specifika krav och målsättningar. Genom att exempelvis diskutera implementeringen av kodförändringar och åtgärder för att anpassa

och förbättra det befintliga systemet får läsaren en djupare förståelse för projektets tekniska aspekter och hur det har utvecklats från början till slut.

I kapitel 4 presenteras de uppnådda resultaten och den färdiga rapporten demonstreras för läsaren. Här ges en översiktlig sammanfattning av projektets framsteg och utfall, och de viktigaste slutsatserna och observationerna lyfts fram. Dessutom inkluderas även en kritisk validering av resultaten och även en validering av de använda metoderna och deras effektivitet i projektet.

Slutligen, i kapitel 5 presenteras en slutledning som inkluderar en sammanfattning av projektet och de uppnådda resultaten, projektets framtid och vidareutveckling efter detta lärdomsprov, samt en kritisk validering av resultaten och även en validering av de använda metoderna och deras effektivitet i projektet.

Genom att strukturera lärdomsprovet på detta sätt ger det en tydlig och logisk överblick över projektets olika faser och aktiviteter. Det möjliggör också för läsaren att följa projektets utveckling från början till slut och att förstå hur varje fas har bidragit till att uppnå de övergripande målen och resultatet.

2 Kravanalys

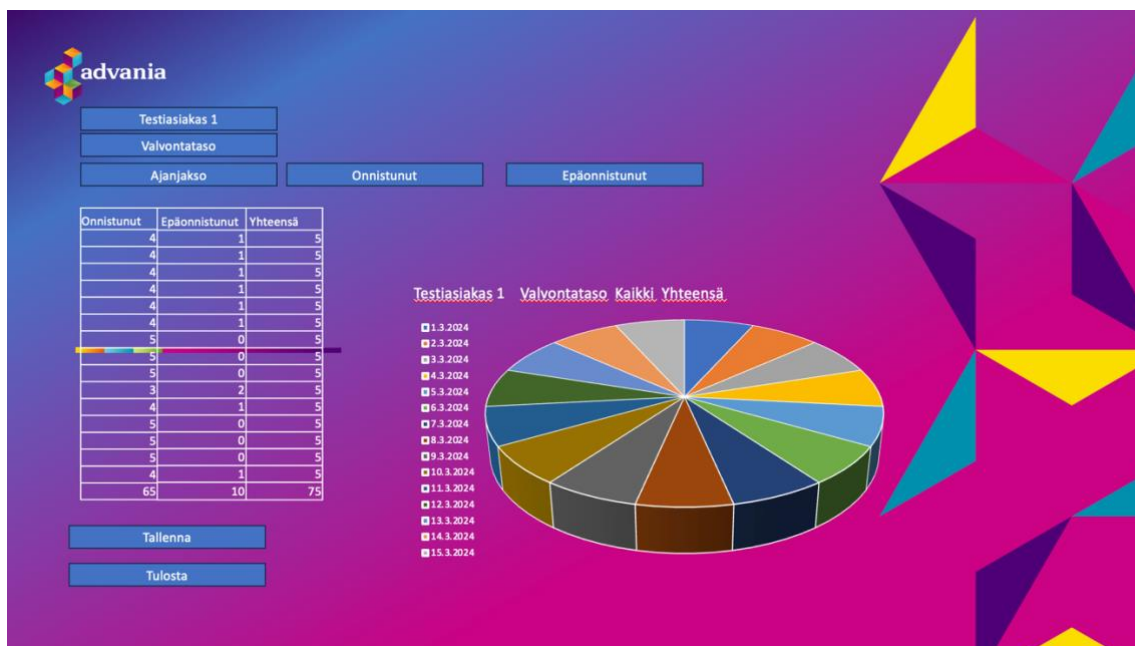
2.1 Definition av kravanalys

Kravanalys är en central metod inom systemutveckling och projektledning med syfte att grundligt förstå och dokumentera kraven för ett system eller produkt (Visure Solutions Inc, 2024). Processen innefattar identifiering, dokumentation, validering och hantering av krav för att säkerställa att de uppfyller användarnas behov och systemets övergripande mål.

För att genomföra en kravanalys är det nödvändigt att börja med att identifiera kraven. Detta görs genom att samla in information från olika intressenter, såsom kunder, användare och andra relevanta parter. Metoder som intervjuer och undersökningar används för att samla in krav och förstå de specifika behoven som behöver adresseras

inför det nya systemet. En viktig del av kravanalysen är valideringen av kraven, där kraven granskas och valideras med alla relevanta intressenter för att säkerställa deras korrekthet, fullständighet och att de verkligen uppfyller användarnas behov. Detta steg är avgörande för att säkerställa att systemet kommer att fungera som avsett och leverera de förväntade resultaten.

Efter att kraven har identifierats och validerats är nästa steg att dokumentera dem på ett strukturerat och tydligt sätt. Detta kan innebära att skapa skisser eller prototyper för att visa hur systemet kommer att fungera och vilken funktionalitet det kommer att erbjuda. I fallet med det nya databasrapporteringsystemet gjordes först en skiss på papper för att visa hur rapporten skulle se ut och vad den skulle innehålla. Prototyp för rapportvalet, dvs vad man vill inkludera i rapporten gjordes också på datorn. Det bestämdes att systemet skulle ge användarna möjlighet att själva välja vilken information de vill inkludera och se i rapporten, utan behov av skraddarsydd rapporter. I figur 3 kan man se prototypen av den grafiska webbversionen för service managers. I figuren visas alltså inte själva rapporten utan här görs valen för vad man vill kunna se i rapporten. Figuren inkluderar drop-down menyer för valen, och på basis av vad man väljer så byggs rapporten upp. I prototypsexemplet syns ”Testiasiakas 1” alla övervakningsnivåer tillsammans, och vilka nivåer som har lyckats och misslyckats under en halv månads tid. Om man fortsatt klickar på ”Tulosta” så printas en version ut, och ifall man klickar på ”Tallenna” så sparas en pdf version på ens lokala dator.



Figur 3. Prototyp av vad man vill att ska synas i själva rapporten.

Efter dokumentationen av kraven är det viktigt att granska den noggrant för att identifiera eventuella brister eller felaktigheter som behöver åtgärdas. Detta säkerställer att kraven är korrekta och fullständiga innan utvecklingen av systemet påbörjas. Genom att följa en strukturerad kravanalysprocess kan man säkerställa att systemet som utvecklas uppfyller användarnas behov och förväntningar på ett effektivt sätt. Detta bidrar till att minimera risken för missförstånd eller felaktigheter under utvecklingsprocessen och säkerställer att den resulterande produkten möter de specifika målen och kraven för projektet.

2.2 Kravanalys på Advanias rapporteringssystem

Kraven för det nya rapporteringssystemet är starkt bundna i behoven och önskemålen från Advanias kunder, vilka länge har efterfrågat mer konkreta genomgångar av databasövervakningen. Dessa genomgångar är avsedda att vara en integrerad del av de månatliga servicemötena mellan service managerna och kunderna, där eventuella önskemål för vidareutveckling av den existerande övervakningen diskuteras.

Det nya rapporteringssystemet har framstått som en av de mest efterfrågade funktionerna från kundernas sida. I service mötena är det även vanligt att gå igenom databaslarmen som har uppstått under den senaste tidsperioden. Under diskussionen

bedöms det ifall det är något som Advania behöver åtgärda eller om det är en fråga om kundens eget agerande som behöver adresseras. Det är en viktig del av kundrelationen att kunna säkerställa en öppen och transparent kommunikation kring övervakningsresultat och eventuella åtgärder. Kraven för det nya systemet strävar mot att betjäna både service managers och kundernas behov. Systemet ska vara lättanvänt och tillgängligt för användarna. Det bör också ge möjlighet för användarna att enkelt filtrera och välja den information de behöver och vill se i rapporterna. Med en webbläsarbaserad version som är kompatibel med alla datorer blir systemet lättillgängligt för alla användare oavsett plattform.

Framtidsplanerna inkluderar att erbjuda rapporten direkt till kunderna så att de själva kan hantera och anpassa data enligt deras behov och preferenser. I nuläget är rapporten avsedd endast för service managers, men en framtida webbläsarbaserad kundversion är under planerad utveckling. Detta ger kunderna möjlighet att själva filtrera och justera data utöver det valda tidsintervallet, vilket ger dem en större autonomi och kontroll över sina databasövervakningar. Kunderna har uttryckt en tydlig önskan om bevis på att deras databaser övervakas i enlighet med avtal, och dessa rapporter kommer att spela en central roll under de månatliga servicemötena mellan Advanias service managers och Advanias kunder. Efter interna diskussioner och planering hos Advania framstod detta projekt som en utmärkt möjlighet att adressera och uppfylla kundernas krav och förväntningar, dvs blev detta projekt ett perfekt projekt för lärdomsprovet.

3 Rapporteringssystemets realisering

3.1 Biblioteksval

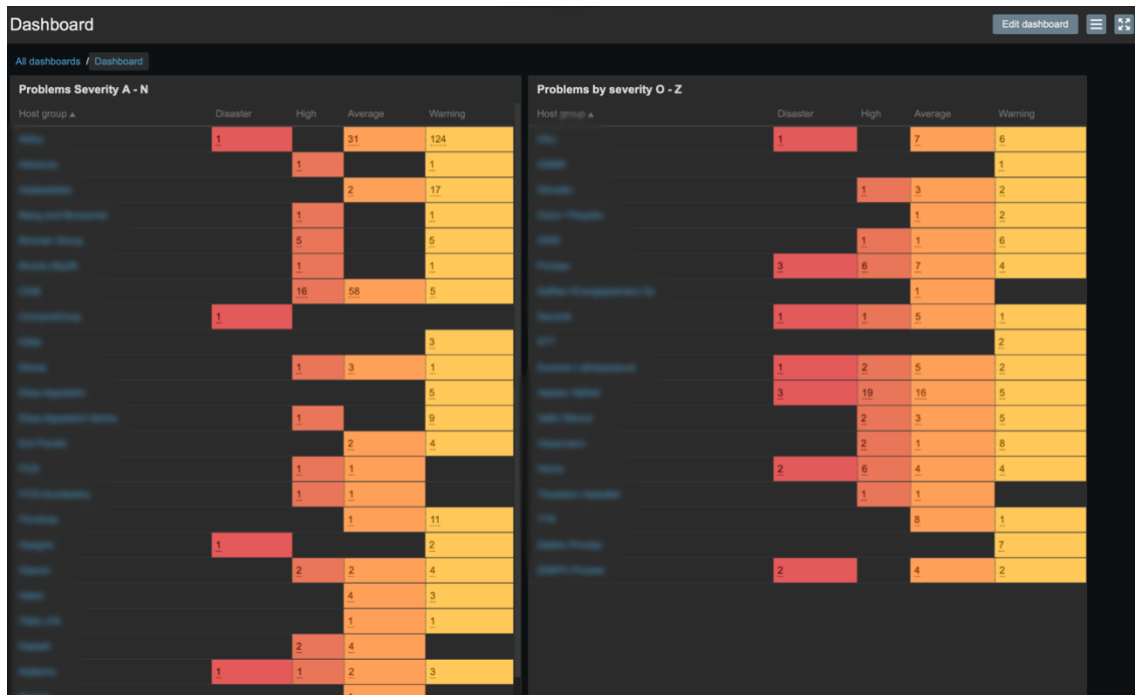
I den första versionen av databasrapporteringssystemet användes flera teknologier och bibliotek, inklusive Zabbix, Power BI (Power BI, 2024) och Python- samt SQL-kod för att utveckla systemet. Advania valde att använda Zabbix-versionen för Linux, som var den senaste versionen för Debian-operativsystemet (Debian, 2024). I figur 4 kan man se hur Zabbix dashboard ser ut. Zabbix installerades lokalt på datorn och genom att vidareutveckla Python-koden i Zabbix kunde den nya versionen av

databasrapporteringsystemet utvecklas och integreras. Under utvecklingen av det nya databasrapporteringsystemet uppgaderades även versionen av Zabbix för att stödja de nya funktionerna och förbättringarna som infördes.

Den nya Python-koden som lades till i systemet användes för att skriva skräddarsydda funktioner och logik för att möta de specifika kraven och behoven hos Advania och deras kunder. Genom att använda Python som programmeringsspråk kunde avancerade rapporteringsfunktioner och datahanteringsmetoder effektivt implementeras.

Kombinationen av Zabbix, Power BI och Python- samt SQL-kod möjliggjorde utvecklingen av en omfattande och kraftfull rapporteringslösning som gjorde det möjligt för programmet att övervaka och analysera kunders databaser på ett effektivt sätt.

Denna integrerade teknologiska strategi gav Advania flexibilitet och skalbarhet för att möta sina kunders behov och förväntningar när det gäller databasövervakning och rapportering.



Figur 4. Zabbix startsida, "dashboard". Här ser man alla kunders övervakning samt övervakningens status med färger i allvarlighetsgrad.

3.2 Python

Python är ett mångsidigt och kraftfullt programmeringsspråk som utvecklades för första gången av Guido van Rossum och släpptes år 1991 (Python Software Foundation, 2024). Det var utformat med en betoning på läsbarhet och enkelhet, vilket gör det till ett populärt val bland både nybörjare och erfarna utvecklare. Under åren har Python vuxit till ett av de mest använda programmeringsspråken i världen, med en stor och aktiv gemenskap av utvecklare som bidrar till dess ständiga utveckling och förbättring. Ett av dess mest kända drag är dess tydliga och lättlästa syntax, vilket gör det lätt att lära sig och använda för att lösa olika problem.

Python används inom en mängd olika områden, från webbutveckling och datavetenskap till artificiell intelligens och maskininlärning. Pythons omfattande bibliotek och ramverk gör det till ett kraftfullt verktyg för att bygga allt från enkla skript till komplexa applikationer.

I detta lärdomsprov har den största delen av projektets utveckling skett med hjälp av Python kod. Genom att skriva och integrera ny kod i Zabbix kunde Advanias krav för övervakning mötas och det nya vidareutvecklade systemets utveckling främjas. Valet av Python som huvudsakligt programmeringsspråk var välgrundat, delvis på grund av Advanias befintliga erfarenhet och kompetens inom Python-utveckling. Genom att redan ha ett etablerat team av Python-utvecklare har utvecklingen kunnat hanteras internt utan att behöva ta hjälp av externa konsultföretag eller specialister.

Denna strategi med intern utveckling har inte bara varit kostnadseffektiv utan också främjat en mer flexibel utvecklingsprocess för Advania. Genom att ha kontroll över hela utvecklingsprocessen och snabbt kunna anpassa och iterera kodbasen efter behov har Advania kunnat reagera dynamiskt på förändrade krav och förväntningar från sina kunder. Detta har också möjliggjort en snabbare och smidigare uppkomst av nya funktioner och förbättringar. Därför är det inte förvånande att utvecklingen av det nya förbättrade databasrapporteringssystemet till stor del är baserat på Python. Den flexibilitet, kraft och skalbarhet som Python erbjuder har varit en central faktor för att

möjliggöra framgången för detta projekt och dess förmåga att uppfylla Advanias och deras kunders behov på ett effektivt sätt.

3.3 SQL

SQL, vilket står för Structured Query Language, används som ett kraftfullt verktyg för att hantera och manipulera data i relationella databaser (Oracle, 2024). Det utvecklades för första gången på 1970-talet av IBM, och har sedan dess blivit en standard inom databashanteringssystem. Med SQL kan användare utföra olika operationer på databaser, såsom att hämta data, att lägga till nya poster, att uppdatera befintliga poster och ta bort poster. Dess kraft ligger i dess förmåga att använda enkla och läsbara kommandon för att interagera med databaser och extrahera den information som behövs.

SQL består av flera olika typer av kommandon, inklusive SELECT för att hämta data, INSERT för att lägga till nya poster, UPDATE för att uppdatera befintliga poster och DELETE för att ta bort poster. Dessutom kan SQL användas för att skapa och hantera tabeller, index och vyer i en relationell databas. SQL är en avgörande färdighet inom datavetenskap och databashantering, och dess användningsområden sträcker sig över en mängd olika branscher och tillämpningar. Med kunskap om SQL kan användare effektivt hantera och analysera stora mängder data, vilket gör det till ett oersättligt verktyg.

SQL syntax har spelat en central roll i detta projekt för Advanias databasövervakning. Genom användning av SQL har projektet möjliggjort flera viktiga funktioner, inklusive övervakning av lyckade eller misslyckade säkerhetskopieringar av databaser. Denna övervakning är kritisk för att säkerställa att data säkerhetskopieras korrekt och att eventuella fel eller avvikelser upptäcks och hanteras snabbt och effektivt. Resultatet av säkerhetskopieringsprocessen, vare sig den varit lyckad eller misslyckad, syns i den nya rapporten som genereras, vilket ger insikter och översikt över databasens status och prestanda. Utöver säkerhetskopieringsövervakningen kan SQL användas för att övervaka och analysera olika aspekter av databasens hälsa och prestanda. Det kan omfatta övervakning av systembelastning, indexhantering, lagringsutrymme och mycket

mer. Genom att använda SQL-queries kan Advania få insikt i databasens tillstånd och identifiera eventuella problem eller potentiella hinder i systemet.

I Zabbix, den plattform som används för övervakning av databaser, stöder Advania också övervakning av flera olika typer av databaser, vilket inkluderar Microsoft SQL Server, MariaDB (MariaDB, 2024), MySQL (MySQL, 2024), PostgreSQL (PostgreSQL, 2024), MongoDB (MongoDB, 2024), och IBM DB2 (IBM, 2024). Denna flexibilitet möjliggör att Advania kan erbjuda sina kunder en omfattande övervakningslösning som kan anpassas efter deras specifika behov och infrastruktur.

Sammanfattningsvis har SQL varit en kritisk komponent i projektet för Advanias databasövervakning, och dess användning har gjort det möjligt att effektivt övervaka och hantera olika aspekter av databasens prestanda och hälsa. Genom att utnyttja SQL kan Advania säkerställa att deras kunders datahanteringssystem fungerar optimalt och att eventuella problem upptäcks och åtgärdas i tid.

3.4 Power BI

Power BI, utvecklat av Microsoft Corporation, är en kraftfull plattform som används för att visualisera och analysera data på ett intuitivt och effektivt sätt. Det erbjuder användarna möjligheten att ansluta till olika datakällor, förbereda och transformera data och skapa interaktiva rapporter och instrumentpaneler för att få insikter och fatta datadrivna beslut.

Med Power BI kan användare enkelt importera data från olika källor såsom Excel-ark, SQL-databaser, molnlagringstjänster och många andra, vilket ger en omfattande vy över företagets data. En av de främsta styrkorna med Power BI är dess möjlighet att skapa dynamiska och interaktiva instrumentpaneler och rapporter. Användare kan skapa anpassade visualiseringar och diagram för att kommunicera data på ett övertygande sätt och upptäcka mönster, trender och insikter som kan vara avgörande för företagets framgång. Dessutom erbjuder Power BI avancerade analysfunktioner som möjliggör prediktiv analys, maskininlärning och affärsintelligens i realtid. Detta gör det möjligt för

användare att dra nytta av de senaste teknikerna inom dataanalys för att få ännu djupare insikter och förutsägelser för framtiden (Microsoft, 2024).

Slutliga rapporten och den grafiska layouten kommer från Power BI. Det är denna plattform som möjliggör användningen av rapporteringen och ger användarna verktyg för att utforska och förstå data på ett djupare sätt. Valet av Power BI som rapporteringsverktyg grundades på flera faktorer, varav en viktig del var dess förmåga att integreras sömlöst med Zabbix serverns övervakningsdatabas. Genom denna direktkoppling kan data enkelt och effektivt extraheras och användas för att skapa mer detaljerade och anpassningsbara rapporter.

En annan fördel med att använda Power BI är dess användarvänliga gränssnitt och möjlighet till grafisk interaktivitet. Användare kan dynamiskt utforska och visualisera data, vilket gör det lättare att upptäcka trender, mönster och avvikelser i datamängden. Det grafiska användargränssnittet gör det också enkelt för både interna och externa användare att förstå och dra nytta av rapporterna utan att behöva djupgående teknisk kunskap.

Genom att använda Power BI kan Advania skapa rapporter som inte bara är informativa utan också visuellt tilltalande och lättförståeliga. Dess flexibilitet och anpassningsbarhet gör det möjligt att skapa skräddarsydda rapporter som passar olika användares behov och preferenser. Med hjälp av Power BI:s funktionalitet kan Advania erbjuda sina kunder en kraftfull och användarvänlig rapporteringslösning som hjälper dem att ta datadrivna beslut och maximera värdet av sina dataresurser.

3.5 Linux

Linux är ett operativsystem som utvecklades av Linus Torvalds och först släpptes år 1991 (Linux Foundation, 2024). Linux har blivit en av de mest populära plattformarna inom datavetenskap och IT. Linux skiljer sig från andra operativsystem genom att det är öppen källkod, vilket innebär att dess källkod är fritt tillgänglig för alla att använda, studera, modifiera och distribuera enligt ens behov.

En av de främsta fördelarna med Linux är dess stabilitet och tillförlitlighet. Genom att ha en solid grund i UNIX-design principer har Linux visat sig vara extremt kraftig och väl lämpad för både servrar och inbyggda system. Dess modulära och skalbara arkitektur gör det möjligt att anpassa och optimera systemet för en mängd olika användningsområden och krav. Linux erbjuder också en mängd olika distributioner eller "distros", vilket ger användare olika val och flexibilitet beroende på deras preferenser och krav (Linux Foundation, 2024). Oavsett om det är för serverdrift, personlig datoranvändning eller inbyggda system finns det en Linux-distribution som passar behoven för alla användare.

Linux operativsystemet valdes på grund av flera fördelaktiga egenskaper. För det första är Linux en gratis klon av Unix-operativsystemet, vilket innebär att det är kostnadseffektivt att använda. Detta är särskilt fördelaktigt för företag och organisationer som strävar efter att maximera sina resurser och minimera sina kostnader.

Unix-operativsystemet är känt för sin stabilitet och tillförlitlighet, vilket är ytterligare en anledning till att Linux valdes. Genom att bygga på samma grundläggande principer som Unix kan Linux erbjuda en liknande nivå av stabilitet och prestanda, vilket är avgörande för ett övervakningssystem som Zabbix.

Eftersom Zabbix är en open-source-plattform, är det naturligt att använda både Linux och Zabbix tillsammans. Linux, med sin öppna källkodsmodell, passar perfekt för att driva och hantera Zabbix-servern. Det ger användare fullständig kontroll och flexibilitet över sitt övervakningssystem. Dessutom ger Linux en bättre prestanda för övervakningssystemet jämfört med alternativ som Windows Server, eftersom Linux är baserat på Unix-operativsystemet och är optimerat för stabilitet och effektivitet.

Genom att välja Linux som operativsystem för Zabbix-servern kan användare dra nytta av dess pålitlighet, prestanda och kostnadseffektivitet. Dess flexibilitet och anpassningsbarhet gör det till ett idealiskt val för att stödja och driva övervakningssystem och andra kritiska IT-tjänster.

3.6 Den tekniska utvecklingen av rapporteringssystemet

Efter att ha valt programvara, tekniker och teknologier samt definierat projektets syfte och önskade resultat, gick man vidare till att välja resurser för utvecklingen av projektet. Det behövdes experter inom Zabbix och SQL för databasövervakning, Power BI och gränssnittsdesign för rapportgenerering, samt kompetens inom Python, Linux och grafisk planering för att säkerställa att systemet fungerade smidigt och användarvänligt. För att säkerställa tillgång till dessa resurser hölls en intern kickoff där projektet presenterades för förvaltningsenheter och där man fick deras stöd. Därefter skapades en prototyp av projektet i Monday.com. Projektet bröts ner i mindre delar och där tilldelades varje del av projektet till de relevanta resurserna för att effektivisera arbetet.

Vidare ordnades ett internt möte med alla resurser för att gå igenom projektet i sin helhet, beskriva användningen av Monday.com för uppföljning och resursallokering samt diskutera projektspecifika detaljer. Där bestämdes även ordningen och ansvarsfördelningen av arbetsuppgifterna och detta dokumenterades i Monday.com för att underlätta uppföljningen av projektets framsteg. Eftersom jag inte kände till vissa av dessa bibliotek, introducerades jag till dem alla via experter inom områdena. Det gav mig möjligheten att lära mig om deras funktioner och användning i projektet. Denna introduktion var avgörande för att kunna förstå hur varje bibliotek skulle integreras och utnyttjas på bästa sätt för att uppfylla projektets krav och mål.

För att säkerställa en smidig kommunikation och samarbete hölls dagliga 15-minutersmöten där personerna inom projektet rapporterade om framsteg och eventuella problem som behövde lösas. Även veckovisa möten hölls för att sammanfatta veckans arbete och planera för kommande vecka, samt månatliga möten för att summera månaden och göra presentationer för projektgruppen. Vid alla möten erbjöds även hjälp inom utvecklingen från de olika parterna.

I min roll i projektet hade jag flera viktiga ansvarsområden som samordnare för arbetet. Jag var den som tog hand om kravanalysen, intervjuerna och den som i stort sätt planerade rapporteringssystemets utséende på basis av Advania Finlands krav. Genom att noggrant följa upp och hantera projektets framsteg i Monday.com spelade jag en central roll i att säkerställa att arbetet fortskred enligt plan och att alla

projektmedlemmar var informerade om status och eventuella utmaningar. Det innebar att jag regelbundet deltog i möten för att diskutera projektets utveckling och eventuella problem som uppstod längs vägen.

En annan viktig del av min roll var att ansvara för att rapportera projektets framsteg och eventuella problem till projektgruppen och andra intressenter. Det var avgörande för att hålla alla involverade parter informerade och för att säkerställa att projektet fortskred enligt plan och att eventuella utmaningar hanterades på ett effektivt sätt.

Utöver samordningsuppgifterna spelade jag också en central roll i utvecklingen av Python-koden för rapporteringssystemet. Jag hade som uppgift att ta hand om normaliseringen av datan till själva rapporten. Den data som originellt skulle vidare till Zabbix kunde Zabbix inte läsa och förstå på rätt sätt, och därav inte generera rapporter eftersom datan inte kunde läsas. Jag var alltså tvungen att vidareutveckla koden och skapa hjälpfunktioner för normaliseringen av data. Jag fick en s.k. stubb kod, vilket var en kod som jag skulle arbeta vidare på. Stubb koden har Advania haft i användning en längre tid, och det stubb koden till en början gjorde var att den granskade att koden är i enlighet med en tillåten data typ i koden, och sedan returnerades antingen data eller ett felmeddelande. Det märktes att några av de använda databasbiblioteken returnerade data i valfritt format, och att data i databasen delvist var rå, dvs. inte normaliserad, och det ledde till problem och därav måste den koden vidareutvecklas. För att normalisera datat så att den passade Zabbix-serverns förväntade format, implementerade jag en intern hjälpfunktion. Denna funktion tog tre parametrar: en referens till det aktuella objektet, datat som skulle normaliseras och den förväntade typen av datavärdet enligt Zabbix-specifikationen. För att utföra normaliseringen använde jag en serie kontroller och omvandlingar baserat på den angivna typ-värdet. Om typ-värdet indikerar att datat ska vara ett numeriskt flyttal, kontrollerade jag för olika datatyper och konverterade dem till float. Om typ-värdet indikerar att datat ska vara ett heltal, konverterade jag den till int. För alla andra typer av värden behandlades datat som en sträng. Under normaliseringsprocessen hanterade jag även eventuella fel som kunde uppstå. Om datat inte matchar den förväntade typen kastas ett felmeddelande med tydlig information om vilken typ av data som förväntades och vilken typ som faktiskt erhöles. Eventuella fel loggades med felsöks-loggning för att underlätta felsökningen. Genom att implementera

denna normaliseringsfunktion möjliggjorde jag korrekt och enhetlig formatering av data innan den skickas till Zabbix-servern för rapportering. Detta säkerställer att datat uppfyller Zabbix-specifikationen och att eventuella fel hanteras på ett robust sätt.

Den kod jag har skapat utgör en viktig del av den större programstrukturen för att hantera data i Zabbix-systemet. Normaliseringsfunktionen är ansvarig för att förbereda och konvertera data i rätt format innan det skickas till Zabbix-servern för rapportering. Det innebar att jag arbetade tillsammans med en Python expert som hjälpte och guidade mig i kodningen, för att säkerställa att koden uppfyllde projektets krav och behov och integrerades sömlöst i den övergripande systemarkitekturen. Koden granskades av Python experten alltid med jämna mellanrum, för att undvika fel och problem. Min utvecklade kod får inte publiceras p.g.a. företagets immaterialrätt. Koden för databasrapporteringsystemet vidareutvecklades i egna filer, där både Python-kod och SQL-kod bearbetades för att möta projektets krav. Dessa filer kallades från Zabbix, och resultatet från körda utvecklade koden syntes sedan i Zabbix. Detta ledde till den lyckade vidareutvecklingen av rapporten och till att den kunde uppfylla de önskade funktionerna och kraven.

Mot slutet av projektet fanns fortfarande behov av att förbättra den visuella presentationen av rapporten, och det var här användningen av Power BI kom in i bilden. Genom att sammanföra Zabbix med Power BI kunde man generera den nya visuella rapporten för Advania Finland, vilket gav användarna en mer interaktiv och estetiskt tilltalande rapportupplevelse.

4 Resultat

4.1 Slutliga rapporten

Både det nya utvecklade rapporteringssystemet och det gamla systemet är baserade på Zabbix-plattformen. För att bättre kunna uppfylla Advanias databaskunders önskemål och möta projektets krav har den vanliga Zabbix open-source-versionen omskrivits med Python. Denna omskrivning har möjliggjort en mer skräddarsydd och avancerad datainsamling från kundernas databasövervakningssystem.

I den nya versionen integreras den omskrivna och vidareutvecklade Python-koden tillsammans med Zabbix, gränssnitt, Power BI samt Python- och SQL-kod för att skapa en mer modern och anpassningsbar rapporteringslösning. Genom att kombinera dessa teknologier möjliggörs en grafisk Power BI översikt som kan anpassas efter användarens behov och preferenser. Användarna har möjlighet att välja vilken typ av information de vill extrahera från databasen och definiera vilken data som ska synas i rapporten. Dessutom kan rapporten filtreras tidsbaserat för att visa data över olika tidsperioder, såsom dagar, veckor, månader eller år. Genom att använda olika typer av diagram, som stapeldiagram, cirkeldiagram eller linjediagram, kan användarna visualisera data på ett lättförståeligt sätt beroende på deras data.

Den nya rapporten erbjuder även fördelen att övervaka säkerhetskopieringar av olika typer av databaser, inklusive Oracle, SQL Server, MariaDB, MongoDB, PostgreSQL och DB2. Dessutom kan rapporten användas av alla service managers för att anpassa och uppfylla sina kunders specifika behov. Rapporten kan integreras i deras månatliga kundmöten för att visa uppdaterad och relevant data. Det nya rapporteringssystemet fungerar också som en intern utvecklingsresurs genom att det möjliggör identifiering och lösning av återkommande problem. Genom att spegla problem mot olika databasversioner kan man hitta lösningar som gynnar flera kunder samtidigt. Detta gör rapporteringssystemet till en viktig del av Advanias utvecklingsprocess.

Det är värt att nämna att även om den nuvarande versionen av rapporteringssystemet har utvecklats och implementerats, så är slutprodukten inte ännu helt klar. Arbete pågår fortfarande för att förbättra och vidareutveckla systemet, och bilden som visas representerar den aktuella versionen av rapporteringssystemet.

I figuren nedan (Figur 5) kan man se en egengjord klon av slutprodukten. Denna klon föreställer själva rapporten. Den originella versionen får inte publiceras p.g.a. företagets immaterialrätt. Kundens information får inte publiceras, därav har det gjorts en egengjord klon av den slutliga rapporten.

fungerande finns det fortfarande utrymme för förbättringar och justeringar i den grafiska layouten och utseendet på rapporterna, för att säkerställa en optimal användarupplevelse. Under testningsfasen visade produkten positiva resultat och god funktionalitet, även om den ännu inte var fullständigt färdigställd. Testningarna genomfördes av Advania Finlands produktutvecklingsavdelning för att säkerställa att produkten uppfyllde de förväntade kraven och standarderna innan den släpptes för användning.

I framtiden förväntas kunderna få ökad autonomi genom att få tillgång till en kundversion av rapporten. Detta innebär att de kommer att kunna filtrera och anpassa rapporten efter sina egna behov och önskemål. Denna förbättring av rapportåtkomst ger kunderna större kontroll och flexibilitet över hur de använder och tolkar sina data. Detta förväntas öka kundtillfredsställelsen och skapa en mer anpassad och engagerande användarupplevelse. Förutom den initiala utvecklingen av rapporten förväntas det även att framtida önskemål från kunderna kommer att leda till kontinuerlig förbättring och utveckling av rapporten. Detta kan inkludera mer detaljerade övervakningsmöjligheter och integrering av nya funktioner och datakällor för att möta kundernas föränderliga behov och förväntningar. Till exempel kan molnbaserade databaser bli en viktigare övervakningsutvecklingsmöjlighet i framtiden, vilket kräver anpassningar och tillägg i rapporten för att kunna hantera dessa nya teknologier effektivt.

En annan potentiell utvecklingsväg är användningen av artificiell intelligens, dvs. AI, för att förbättra dataanalysen och upptäcka mönster och avvikelser i datamängderna. Genom att integrera AI-tekniker kan systemet automatiskt identifiera och analysera potentiella fel eller avvikelser i övervakningsdata och föreslå åtgärder för att lösa problemen. Detta skulle göra övervakningsprocessen mer effektiv, vilket i sin tur leder till förbättrad prestanda och tillförlitlighet i systemet över tid.

Sammanfattningsvis kommer rapportutvecklingen att vara en kontinuerlig process som anpassar sig till kundernas behov och teknologiska framsteg. Genom att integrera kundfeedback, nya teknologier såsom molnbaserade databaser och AI, förväntas rapporten fortsätta att utvecklas och förbättras för att möta framtidens krav och förväntningar från både Advanias kunder och marknaden i stort.

Vid genomförandet av projekt med open-source tekniker är det ytterst viktigt att vara medveten om de potentiella utmaningarna som kan uppstå. Användningen av open-source program innebär att man noga måste överväga hur omskrivningar av filer och utvidgning av systemet påverkar prestandan och stabiliteten hos programmet. Det är lätt att göra open-source program långsamma med vissa förändringar, vilket kan hindra projektets framsteg och leda till oönskade resultat. Därför är det av stor betydelse att ha en god förståelse för open-source principer och att vara medveten om hur man bäst kan optimera och anpassa systemet för att möta projektets krav.

I fallet med ett internt projekt där projektet är fördelat mellan resurserna, men där det dagliga arbetet har prioritet, är det ytterst viktigt att noggrant planera och hantera tidtabeller. Kundrelaterade åtaganden har ofta företräde och kan påverka projektets tidslinje och resursallokering. Därför är det viktigt att ha flexibilitet och förmåga att anpassa sig till förändringar i prioriteringar och scheman för att säkerställa projektets framgång. Det är viktigt att spara tillräckligt med tid för projekt som detta, eftersom hjälp kanske inte alltid finns att få direkt.

En fördel med att bygga vidare på ett befintligt program är att grunden redan finns på plats, vilket minskar behovet av omfattande planering och utveckling från grunden. Det sparar tid och resurser som annars skulle ha behövts för att skapa flödesscheman, utforma algoritmer och välja lämpliga bibliotek. Istället kan man fokusera på att anpassa och utöka den befintliga lösningen för att uppfylla kundens specifika behov och krav. Detta ger en smidigare och mer effektiv utvecklingsprocess, vilket är avgörande för att leverera en högkvalitativ slutprodukt inom rimlig tidsram.

Källor

Advania. (15 april 2024). *Meistä*. <https://www.advania.fi/meista>

Debian. (2024). <https://www.debian.org/>

IBM DB2 (2024) <https://www.ibm.com/products/db2>

Linux Foundation. (2024). <https://www.linuxfoundation.org/>

MariaDB. (2024) <https://mariadb.org/>

Microsoft Power BI (2024) <https://www.microsoft.com/en-us/power-platform/products/power-bi>

Monday. (2024). <https://monday.com/>

MongoDB (2024) <https://www.mongodb.com/>

MySQL (2024) <https://www.mysql.com/>

Oracle. (23 april 2024). SQL for Accessing, Defining, and Maintaining Data. <https://www.oracle.com/database/technologies/appdev/sql.html>

PostgreSQL (2024) <https://www.postgresql.org/>

Python Software Foundation. (2024). <https://www.python.org/psf-landing/>

Visure Solutions Inc. (2024). Kravanalys. <https://visuresolutions.com/sv/sp%C3%A5rbarhetsguide-f%C3%B6r-kravhantering/kravanalys/>

Zabbix. (2024, 15 april). <https://www.zabbix.com/>