Berufliches Schulzentrum für Elektrotechnik Dresden

Fachbereich Informationstechnik

Projektdokumentation

Lernfeld 9 - Projekt 3

Auftraggeber: Doubtful-Joy SE
Auftragnehmer: High-Secure GmbH - Projektteam IT20/2 Gruppe 7
Auftragsdatum: 2021.11.15

Historie:

Inhaltsverzeichnis

Abkürzungsverzeichnis

1 Pflichtenheft **1**

1.1 Auftraggeber und Auftragnehmer . . . 1
1.2 Ausgangslage . . . 1
1.3 Projektziel . . . 1
1.4 Funktionsspezifikation . . . 2
1.5 Datenspezifikation . . . 2
1.6 Schnittstellenspezifikation . . . 4
1.7 Rahmenbedigungen . . . 4
1.8 Qualitätsbetrachtung . . . 5
1.9 Projektplanung . . . 5
1.10 Kosten-Nutzen-Analyse . . . 6

2 Auswertung und Reflexion **7**

2.1 Ablaufdokumentation . . . 7
2.2 Einrichtung IPFire . . . 7
2.3 Einrichtung Admin-PC . . . 8
2.4 Einrichtung DHCP-DNS-DB-Server . . . 9
2.4.1 Einrichtung der CentOS-Installation . . . 9
2.4.2 Einrichtung des DHCP und DNS . . . 9
2.4.3 Einrichtung des Datenbank-Servers . . . 9
2.5 Einrichtung Webserver . . . 12
2.5.1 Einrichtung der CentOS-Installation . . . 12
2.5.2 Einrichtung des Backends . . . 12
2.5.3 Einrichtung des Frontends . . . 12
2.6 Soll-Ist-Vergleich . . . 12
2.7 Abweichung zum Zeitplanung . . . 12
2.8 Optimierungsvorschläge zur Projektrealisierung . . . 13

Abbildungsverzeichnis **14**

Tabellenverzeichnis **15**

Listings **16**

Anhang **17**

Abkürzungsverzeichnis

API Application Programming Interface 12

CRUD create, read, update und delete 12

DB Datenbank 7

DHCP Dynamic Host Configuration Protocol 1

DNS Domain Name System 1

DMZ Demilitarisierte Zone 3

VM virtuelle Machine 7

1 Pflichtenheft

1.1 Auftraggeber und Auftragnehmer

Beim Auftraggeber handelt es sich um die Gaming-Plattform **Doubtful-Joy SE**. Ansprechpartner sind

Tabelle 1.1: Ansprechpartner Auftraggeber

Funktion	Name	Vorname	Email
Auftraggeber	Hempel	Steffen	‹hempel@bszetdd.lernsax.de›

Beim Auftragnehmer handelt es sich um das **High-Secure GmbH - Projektteam IT20/2 Gruppe 7**. Ansprechpartner sind

Tabelle 1.2: Ansprechpartner Auftragnehmer

Funktion	Name	Vorname	Email
Projektmanager	Egermann	Péter	‹i20egermannpe@bszetdd.lernsax.de›
Teamleiter	Leyrer	Johannes	‹i20leyrerjo@bszetdd.lernsax.de›
Netzwerkingenieur	Brethfeld	Vinzenz	‹i20brethfeldvi@bszetdd.lernsax.de›

1.2 Ausgangslage

Die existierende Support-Infrastruktur der Gaming-Plattform Doubtful-Joy SE lässt sich über Mail und Telefon kontaktieren. Dabei wird jeder Anruf und jede Mail individuell von einem Mitarbeiter als Ticket gespeichert und in einem zentralen Laufwerk abgelegt. Effizienz, Ordnung und Übersichtlichkeit sind nicht ausreichend vorhanden.

1.3 Projektziel

Die Gaming-Plattform Doubtful-Joy SE möchte ihre existierende Support-Infrastruktur durch ein Ticketsystem ersetzen. Dieses soll für Kunden und Mitarbeiter über ein Web-Interface erreichbar sein. Tickets sollen über dieses direkt erstellt und mit beliebig vielen Attachments versehen werden können.
Außerdem soll eine Segmentierung der Netzinfrastruktur mit einer sichereren Trennung von öffentlich erreichbaren Diensten und dem Intranet eingerichtet werden. Ebenso sollen die internen Dienste Domain Name System (DNS) und Dynamic Host Configuration Protocol (DHCP) auf einem separatem System bereitgestellt werden, um eine Abhängigkeit von der Firewall auszuschließen.

Doubtful-Joy SE setzt auf RedHat und binärkompatible Systeme, weshalb diese System-Strategie weiterhin umgesetzt werden soll.

1.4 Funktionsspezifikation

Von der Realisierung sind betroffen:

Manware

- Projektteam IT20/2 Gruppe 7
- Support-Mitarbeiter des Auftraggeber
- IT-Mitarbeiter des Autraggebers

Orgware

- Sicherheitsanforderungen
- Benutzerhandbuch
- Benutzerschulung

Hardware

- Server
- Mitarbeiter-PCs

Software

- VM-Ware
- Datenbank-Server
- Web-Server
- Firewall-System
- DNS
- DHCP

1.5 Datenspezifikation

Da von etwa 1000 Telefonanrufen und Emails pro Tag ausgegangen wird, kann dies etwa 1:1 in 1000 Tickets übertragen werden. Der Speicherbedarf pro Ticket wird hier im Schnitt auf etwa 5 MB geschätzt, da wahrscheinlich häufiger Anhänge in Bildform zur besseren Problembeschreibung genutzt werden. Zusätzlich wird davon ausgegangen, dass die Daten zur Sicherheit und Nachvollziehbarkeit für ein Jahr gespeichert werden, wodurch die Datenbank 1830 GB Speicher in einem Jahr benötigt.

$$\frac{5\,\mathrm{MB}}{\mathrm{Ticket}} \cdot \frac{1000\ \mathrm{Ticket}}{\mathrm{Tag}} = \frac{5000\,\mathrm{MB}}{\mathrm{Tag}}$$

$$\frac{5000\,\mathrm{MB}}{\mathrm{Tag}} \cdot 365\mathrm{Tage} = \frac{1\,825\,000\,\mathrm{MB}}{\mathrm{Jahr}} \mathrel{\hat{=}} \frac{1830\,\mathrm{GB}}{\mathrm{Jahr}}$$

Da es keine Good-Practice ist, die Bilder in der Datenbank zu speichern, wird nur der Dateipfad zu den Bildern in der Datenbank hinterlegt, die Bilder selbst liegen auf der Festplatte des Webservers. Damit verringert sich der geschätzte Speicherbedarf der Datenbank auf etwa 183 GB pro Jahr.

$$\frac{0{,}5\,\mathrm{MB}}{\mathrm{Ticket}} \cdot \frac{\mathrm{Ticket}}{\mathrm{Tag}} = \frac{500\,\mathrm{MB}}{\mathrm{Tag}} \mathrel{\hat{=}} \frac{183\,\mathrm{GB}}{\mathrm{Jahr}}$$

Die Bilder selbst benötigen zum aktuellen Stand auf der Festplatte 1643 GB Speicher pro Jahr.

$$\frac{4{,}5\,\mathrm{MB}}{\mathrm{Bild}} \cdot 1000\frac{\mathrm{Bild}}{\mathrm{Tag}} = \frac{4500\,\mathrm{MB}}{\mathrm{Tag}} \mathrel{\hat{=}} \frac{1643\,\mathrm{GB}}{\mathrm{Jahr}}$$

Die Art von Daten sind personenbezogene Daten in Text- und Bildform.
Der Datenfluss geht vom Clienten zur Demilitarisierte Zone (DMZ) und zur Bearbeitung dann zum PC des Support-Mitarbeiters, grafisch dargestellt in Abb. 1.1 auf der nächsten Seite.

1.6 Schnittstellenspezifikation

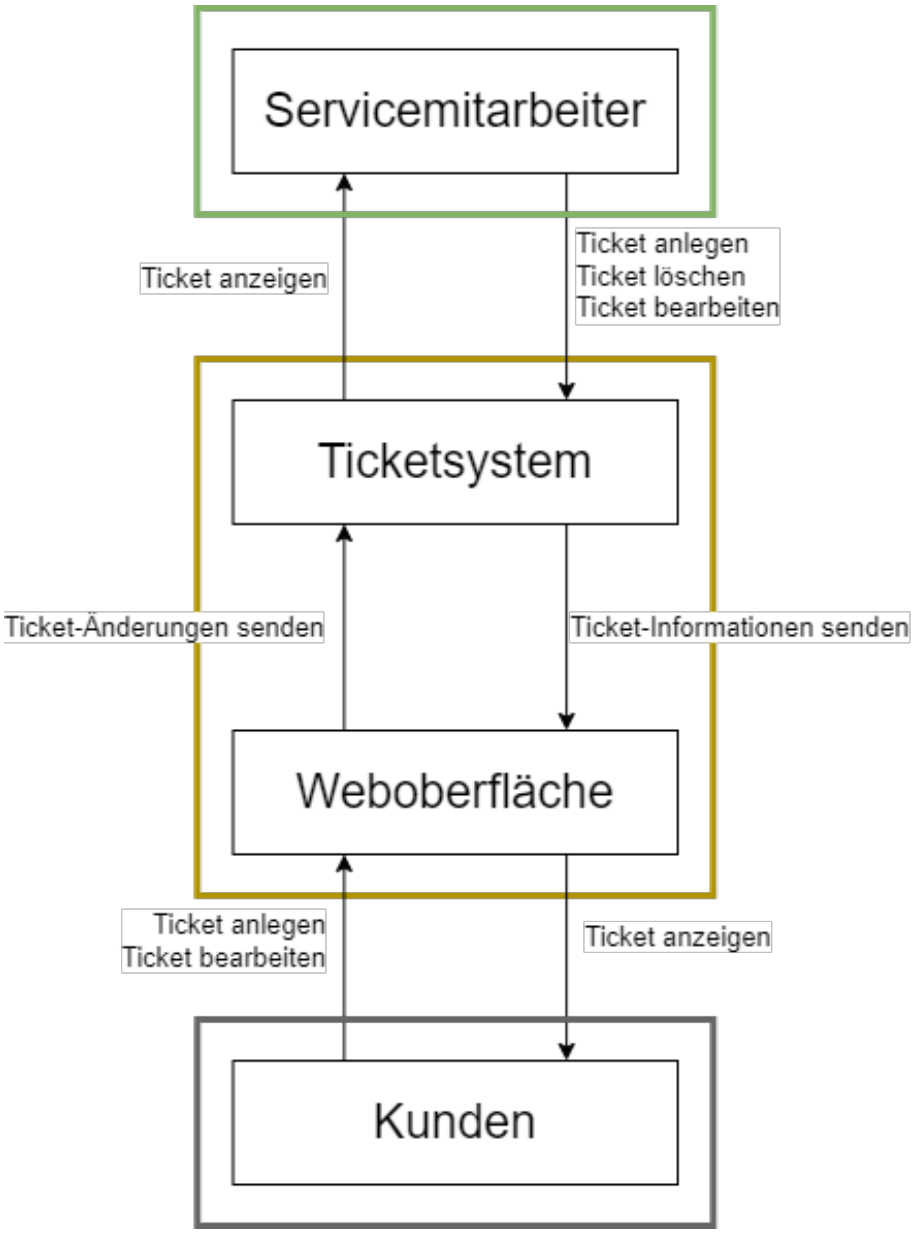


Abbildung 1.1: Schnittstellenspezifikation

1.7 Rahmenbedigungen

Der Auftraggeber hat folgende Ressourcen bereitzustellen und Mitwirklungspflichten:

- Server
- Mitarbeiter-PCs
- Zugriff auf alle zu bearbeitenden Systeme und Zutritt zu den notwendigen Räumlichkeiten
- Kooperation und eventuell notwendigen lokalen Support

1.8 Qualitätsbetrachtung

Die Arbeitspakete werden stets während der Bearbeitung sowie nach der Fertigstellung auf Funktion und Qualität überprüft.
Wöchentlich werden Meetings abgehalten um den Stand des Projekts zu erörtern und auf eventuell auftretende Probleme zeitnah reagieren zu können.
Die Zeitplanung und damit der Aufwand ist in Abb. 1.3 auf der nächsten Seite in kleinem Format und groß in Abb. A.1 auf Seite 18 zu sehen. Für einen langfristigen Support für nach der der Fertigstellung wird ein zusätzliches Angebot vorgelegt.

1.9 Projektplanung

Die Projektplanung ist im Projektstrukturplan, zu sehen in Abb. 1.2, und im Gantt-Diagramm, zu sehen in Abb. 1.3 auf der nächsten Seite, bzw. Abb. A.1 auf Seite 18, abgebildet. Ebenso wird der im Anhang Seite 20 zu betrachtende Netzwerkplan Abb. B.1 auf Seite 20 umgesetzt.

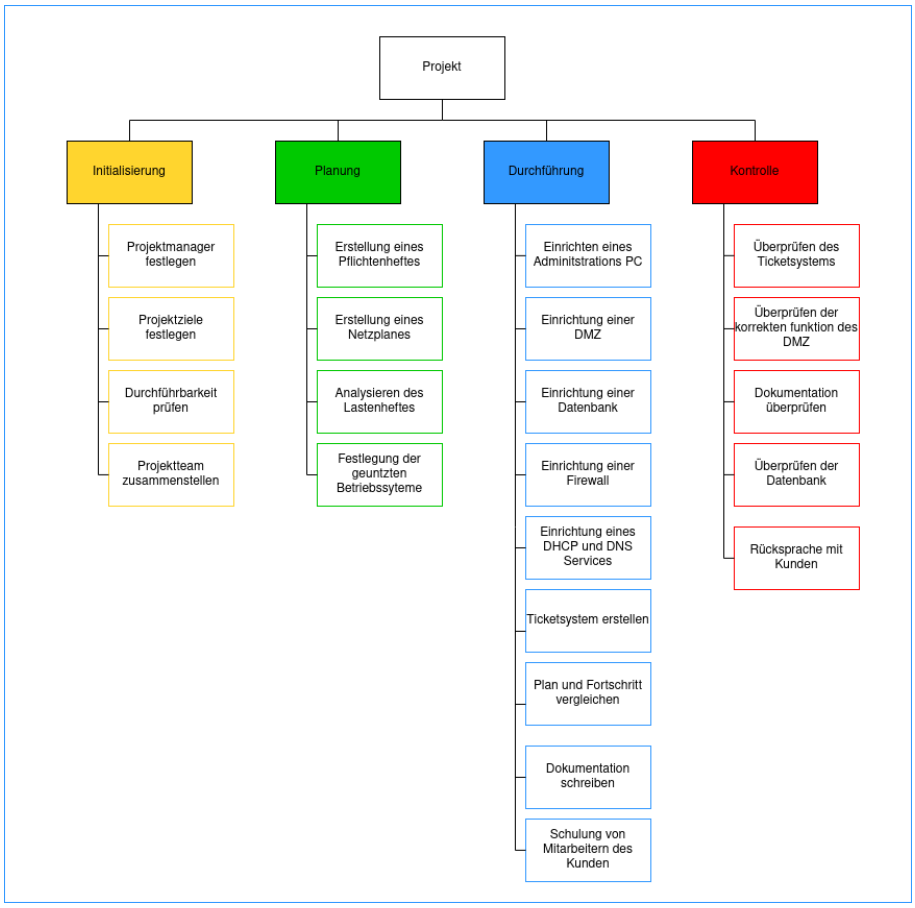


Abbildung 1.2: Projektstrukturplan

GANTT Diagramm Gruppe 7

PROJEKTTITEL	Projekt 3	COMPANY NAME	Projektteam IT20/2 Gruppe 7
PROJEKT MANAGER	Péter Egermann	DATE	11.12.2021

WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1	Projektplanung					87%
1.1	Pflichtenheft erstellen	Peter	15.11.2021	22.12.2021	28	80%
1.2	Projektstruktuplan erstellen	Vinzenz	15.11.2021	22.12.2021	28	80%
1.3	Netzplan überarbeiten	Gruppe 7	15.11.2021	22.12.2021	28	100%
2	Durchführung					0%
2.1	Server einrichten (DHCP, DNS)	Vinzenz	03.01.2022	05.01.2022	3	0%
2.2	Firewall einrichten	Vinzenz	03.01.2022	05.01.2022	3	0%
2.3	Netzwerk einrichten	Vinzenz	03.01.2022	05.01.2022	3	0%
2.4	Netzwerk testen	Vinzenz	06.01.2022	06.01.2022	1	0%
2.5	Firewall testen	Vinzenz	06.01.2022	06.01.2022	1	0%
2.6	Datenbankschema erstellen	Peter	03.01.2022	05.01.2022	3	0%
2.7	Datenbank einrichten	Peter	06.01.2022	06.01.2022	1	0%
2.8	Backendgrobplanung / PAP	Johannes	03.01.2022	05.01.2022	3	0%
2.9	Backend erstellen	Johannes	06.01.2022	11.01.2022	4	0%
2.10	Backend testen	Johannes	10.01.2022	12.01.2022	3	0%
2.11	Frontendgrobplanung / PAP	Johannes	03.01.2022	05.01.2022	3	0%
2.12	Frontend erstellen	Johannes	07.01.2022	12.01.2022	4	0%
2.13	Frontend testen	Johannes	11.01.2022	13.01.2022	3	0%
3	Projekt abschliessen					0%
3.1	Benutzerhandbuch	Peter	28.02.2022	02.03.2022	3	0%
3.2	Projektuebergabe	Peter	04.03.2022	04.03.2022	1	0%

Legende

UW3	KW46	15.11.2021	19.11.2021	
UW4	KW47	22.11.2021	26.11.2021	Meilenstein Phase 1
UW5	KW1	03.01.2022	07.01.2022	
UW6	KW2	10.01.2022	14.01.2022	Meilenstein Phase 2
UW7	KW9	28.02.2022	04.03.2022	Meilenstein Projektabgabe

Abbildung 1.3: Gantt-Diagramm

1.10 Kosten-Nutzen-Analyse

Eine Kosten-Nutzen-Analyse ist zum jetzigen Zeitpunkt nicht notwendig, da der Support erst mal entlastet werden muss. Dies ist durch das neue System auf jeden Fall der Fall, da quasi der Kunde das Ticket erstellt und nicht der Support-Mitarbeiter. Somit kann sich voll auf das Beheben des Problems konzentriert werden.

2 Auswertung und Reflexion

2.1 Ablaufdokumentation

In diesem Projekt wurden drei Netze eingerichtet, um ein echtes Netzwerk wiederzugeben. Dabei handelt es sich um das rote Netz, das das Internet wiedergeben soll, das orange Netz, was eine sogenannte DMZ darstellt und das grüne Netz, das das interne Netz darstellt. Um die Kommunikation und den Zugriff zwischen den Netzen zu regeln wird in diesem Projekt die Firewall verwendet. Somit kann aus dem roten Netz nur mit dem orangen Netz kommunizieren werden, aus dem grünen Netz ist kein Zugriff auf das Internet möglich und zwischen dem orangen und grünen Netz sind nur bestimmte Ports zur Kommunikation und Datenübertragung zugelassen.
Um Maschinen in den verschiedenen Netzen darzustellen wurden vier verschiedene virtuelle Machinen (VMs) aufgesetzt, die bis auf der IPFire auf CentOS 8 Stream basieren:

- IPFire (als Knotenpunkt für alle drei Netze)
- Admin-PC (grünes Netz)
- DHCP-DNS-Datenbank (DB)-Server (grünes Netz)
- Webserver (DMZ, oranges Netz)

2.2 Einrichtung IPFire

Vor dem Einrichten der IPFire müssen noch zwei Netzwerke zu dem schon bestehenden Netzwerk hinzugefügt werden, da die Firewall mit drei verschiedenen Netzen interagieren soll. Noch dazu werden den einzelnen Netzen verschiedene MAC-Adressen zugeteilt, damit sie in der späteren Nutzung zuordenbar sind.
In unserem Projekt wurden die Netzwerke wie in Tabelle 2.1 zu sehen verteilt.

Tabelle 2.1: Netzwerkauslegung

Netzwerk-Farbe	MAC-Adresse	Netzwerk
Rot	00:50:56:32:BA:0F	NAT
Grün	00:50:56:3D:EC:D6	VMnet1
Orange	00:50:56:3E:56:B7	VMnet2

Als Hostname der Firewall wurde ïpfireünd als Domaine wurde "doubtful-joy07.com"festgelegt. Nach dem Auswählen der Sprache wurde aufgrund der Kundenspezifikation das Filesystem ëxt4 Filesystemäusgewählt. Nach dem Zuweisen der einzelnen Netze mit den IP- und MAC-Adressen wurde der DHCP deaktiviert, damit es später mit dem DHCP-Server im grünen Netz nicht zu Komplikationen führt.

Nachdem die Firewall eingerichtet wurde können aus dem grünen Netz mittels des IPFire-WebInterfaces verschiedene Einstellungen der Firewall bearbeitet werden. Für die Verbindungen zwischen Admin-PC, DHCP-DNS-DB-Server und Webserver sowie für die Erreichbarkeit des Webservers aus dem roten Netz wurden vier verschiedene Regeln erstellt, die in n Abb. 2.1 zu sehen sind.

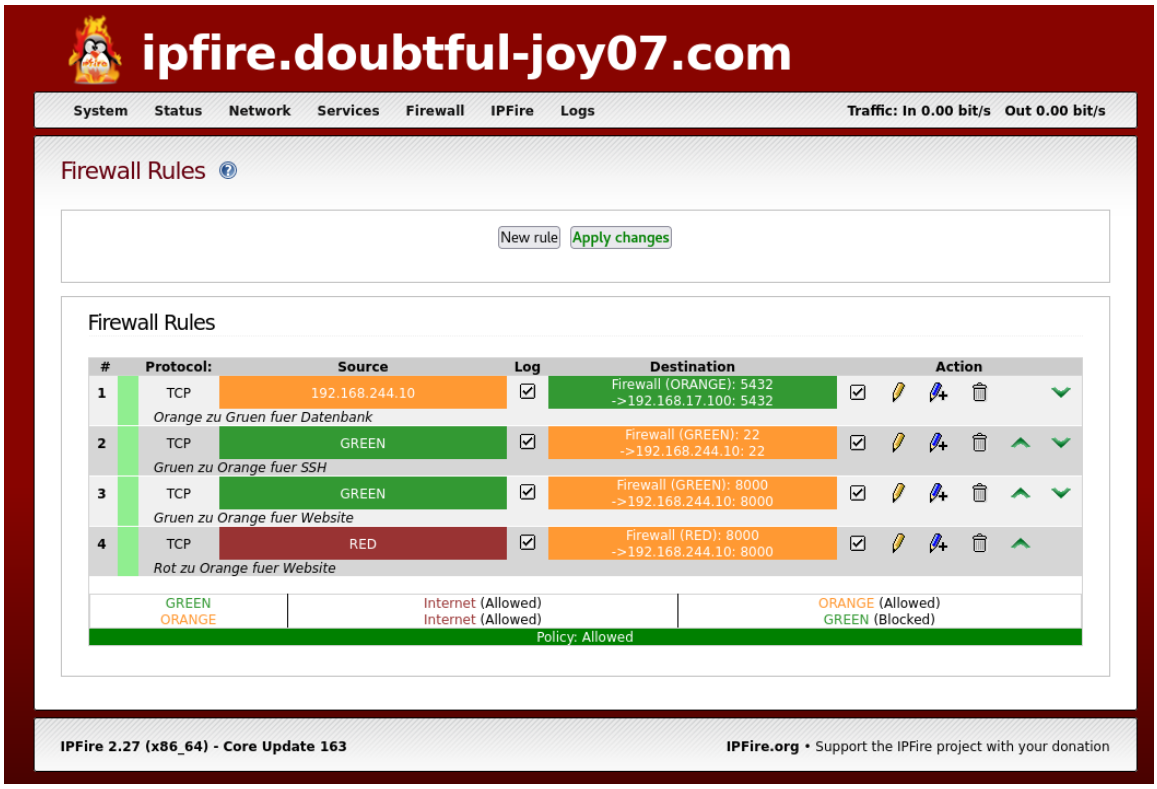


Abbildung 2.1: Firewall-Regeln

Die erste Regel dient der Kommunikation zwischen Webserver (DMZ) und Datenbank (grünes Netz), sodass Tickets abgerufen und gespeichert werden können. Damit aus dem grünen Netz der Webserver gestartet und gestoppt werden kann, wurde Regel 2 implementiert. Die Erreichbarkeit des WebInterface der Ticket-Seite durch die Mitarbeiter aus dem grünen Netz wurde mit Regel 3 erreicht. Die letzte Regel erlaubt den Zugriff aus dem Internet auf die Ticket-Website.

2.3 Einrichtung Admin-PC

Nach der Standard-Installation von CentOS 8 Stream wurden das Netzwerk der VM angepasst. Hier wurde der DNS auf die IP-Adresse des DHCP-DNS-DB-Servers gesetzt.
Da in CentOS 8 Stream SSH-Client und -Server bereits installiert und aktiviert sind, konnte der Webserver direkt angesprochen werden. Dies erfolgte ueber das Gateway des grünen Netzes, wie in Abb. 2.2 auf der nächsten Seite zu sehen. Um zu sehen, ob der Webserver aktiv ist, können mittels `ps -ef | grep python` alle laufenden Python-Anwendungen aufgelistet werden, was ebenfalls in Abb. 2.2 auf der nächsten Seite zu sehen ist.
Ist der Webserver aktiv und soll gestoppt werden, kann mittels `ps -ef | grep python` die ID

Abbildung 2.2: SSH-Login sowie Auflisten aller laufenden Python-Anwendungen

des Scripts ermittelt und mittels `kill -9 ID` gestoppt werden. Dies ist in Abb. 2.3 zu sehen.

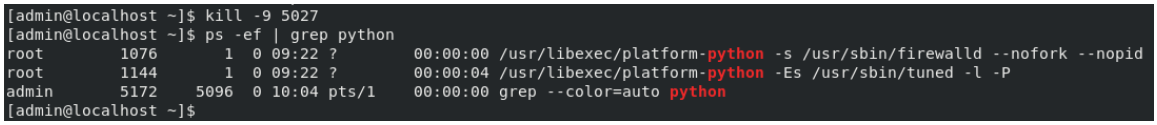


Abbildung 2.3: Stoppen einer bestimmten Python-Anwendungen

Soll der Webserver gestartet werden, kann dies mittels Navigation in den Ordner, in dem die auszuführende Datei liegt und `python3.10 name-der-datei &` gestartet werden, zu sehen in der Abb. 2.4. Das & erlaubt das Laufen der Anwendung im Hintergrund und wird so nicht gestoppt, wenn die SSH-Verbindung geschlossen wird.

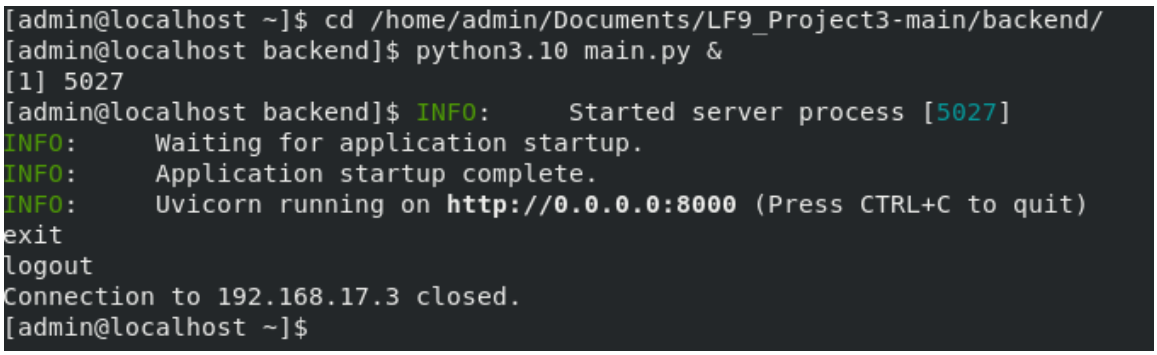


Abbildung 2.4: Starten einer bestimmten Python-Anwendungen

2.4 Einrichtung DHCP-DNS-DB-Server

2.4.1 Einrichtung der CentOS-Installation

2.4.2 Einrichtung des DHCP und DNS

Die Funktion des DNS kann mittels `nslookup`, zu sehen in Abb. 2.5 auf der nächsten Seite, oder ueber das Aufrufen des Frontends des Webservers, zu sehen in Abb. 2.6 auf der nächsten Seite, überprüft werden.

2.4.3 Einrichtung des Datenbank-Servers

Die Einrichtung eines Datenbank-Servers erfolgte mittels der Anleitung von Aaron Kili. [2] Mit dieser wurde PostGreSQL, ein „leistungsstarkes, weit verbreitetes, quelloffenes, plattform-

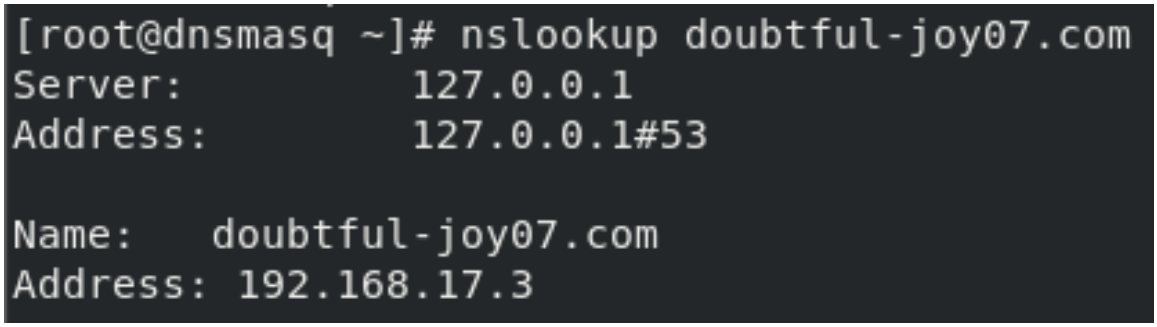


Abbildung 2.5: Überprüfen des DNS mittels Namens

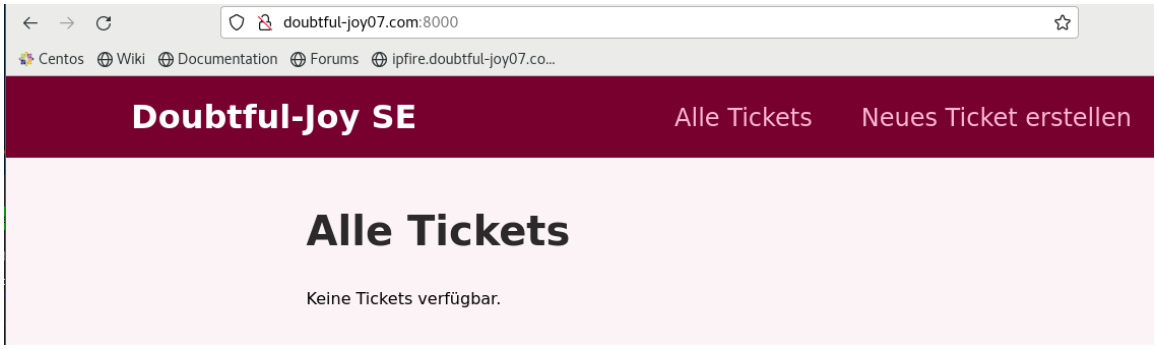


Abbildung 2.6: Aufrufen des Frontends mittels Name

übergreifendes und fortschrittliches objektrelationales Datenbanksystem“ [2] und pgAdmin, ein „fortschrittliches, quelloffenes, voll funktionsfähiges und webbasiertes Verwaltungs- und Managementwerkzeug“ [2], installiert.
Mittels des WebInterfaces, das pgAdmin bereitstellt, lässt sich eine Datenbank erstellen, zu sehen in Abb. 2.7 auf der nächsten Seite. Der Name der Datenbank wurde auf „tickets“, der Benutzername auf „ticketadmin“ und das Passwort auf „adminadmin“ festgelgt.
Um die Kommunikation zwischen der Datenbank und dem Backend des Webservers zu erlauben, muss der Port 5432 freigegeben werden, zu sehen in Listing 2.1.

Listing 2.1: Port-Freigabe einer CentOS-Firewall

```
# firewall-cmd --zone=public --add-port=5432/tcp --permanent
# firewall-cmd --reload
```

Ebenso muss die `postgresql.conf` angepasst werden, so dass ein Zugriff von außerhalb überhaupt möglich ist. [1] Dies ist in folgendem Listing 2.2 zu sehen.

Listing 2.2: Einrichtung Zugriff PostGreSQL

```
# nano /var/lib/pgsql/data/postgresql.conf
listen_addresses = '*'
```

Da im Front- und Backend Timestamps als Zahlen verwendet werden, muss die Tabelle noch geändert werden, damit kein `integer-out-of-range`-Fehler geworfen wird. Dies wurde wie in Listing 2.3 auf Seite 12 umgesetzt.

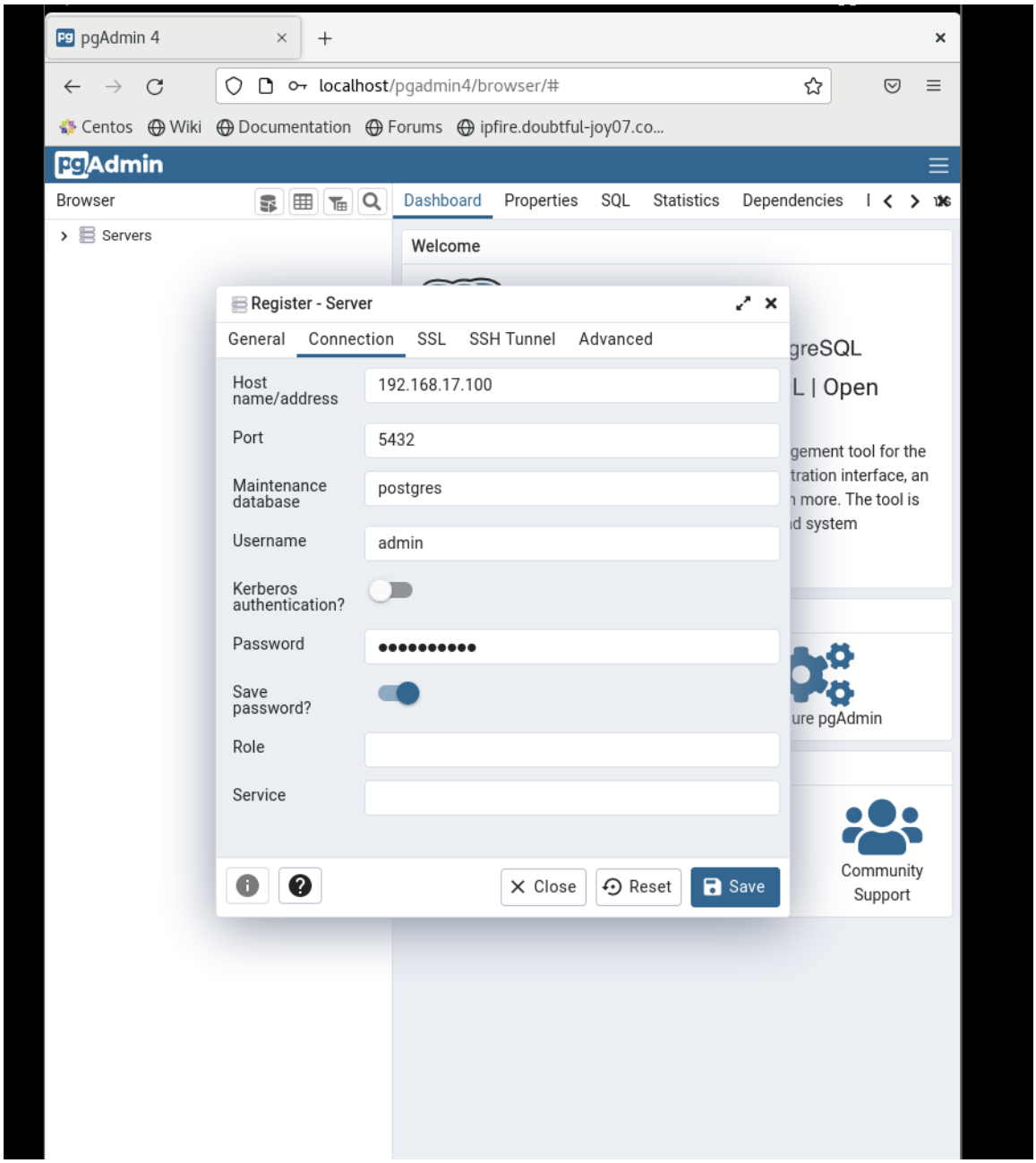


Abbildung 2.7: Einrichtung der Tickets-Datenbank

Listing 2.3: Ändern der tickets-Tabelle

```
ALTER TABLE order_detail ALTER COLUMN amount TYPE BIGINT;
```

2.5 Einrichtung Webserver

2.5.1 Einrichtung der CentOS-Installation

Nach der Standard-Installation von CentOS 8 Stream wurden das Netzwerk der VM angepasst. Hier wurde die IP-Adresse fest auf `192.168.244.10`, die Netzwerkmaske auf `255.255.255.0` und dsa Gateway auf `192.169.244.3` gesetzt. Außerdem wurde der Port 8000 und Port 22 freigeben. Port 8000 dient dem Zugriff auf die Website von den beiden anderen Netzen aus und Port 22 erlaubt den SSH-Zugriff des Admin-Pcs. Als Beispiel ist die Port-Freigabe für Port 8000 im Listing 2.4 zu sehen.

Listing 2.4: Port-Freigabe einer CentOS-Firewall

```
# firewall-cmd --zone=public --add-port=8000/tcp --permanent
# firewall-cmd --reload
```

Abschließend wurde Python3.10 installiert, um das Backend betreiben zu können.

2.5.2 Einrichtung des Backends

Das Backend wurde mit Python3.10 umgesetzt. Hierfür wurde mit Hilfe von FastAPI eine Application Programming Interface (API) geschrieben, die verschiedene Routen bereitstellt, mit denen create, read, update und delete (CRUD)-Anweisungen ausgeführt werden können. Außerdem dient das Backend auch gleich als Server für das Frontend, da es eben dieses bereitstellt. Die benötigten Pakete können mittels `pip3.10 requirements.txt` installiert werden.

2.5.3 Einrichtung des Frontends

Für das Frontend wurde JavaScript-Softwarebibliothek React verwendet. Hier wurden alle vom Kunden geforderten Anzeige und Bedienelemente implementiert. Da das Frontend bereits nach der Umsetzung gebaut wurde und durch das Backend bereitgestellt wird, müssen keine Pakete installiert werden.

2.6 Soll-Ist-Vergleich

Der Zustand der abgelieferten Arbeit entspricht dem Soll-Zustand und somit den Kundenwünschen. Es wurden alle Kriterien umgesetzt. Das Firewall-System, DHCP und DNS, Webserver und Datenbanksystem funktionieren einwandfrei.

2.7 Abweichung zum Zeitplanung

Der ursprüngliche Zeitplan, zu sehen in Abb. 1.3 auf Seite 6, bzw. Abb. A.1 auf Seite 18, konnte nicht eingehalten werden. Es gab zwei Probleme:

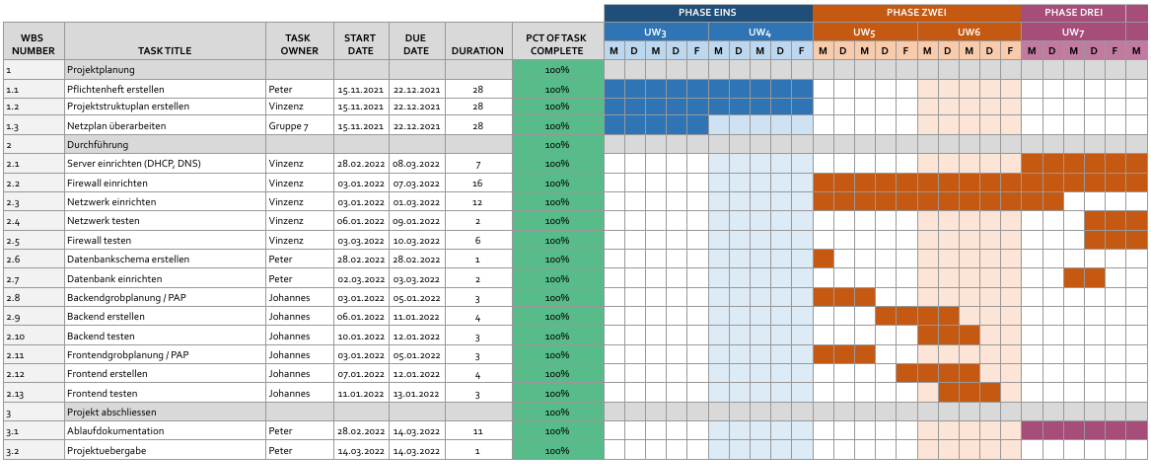
- Die Linux-Clients im IPFire Netz konnten keine Software installieren.
  Hier wurde vermutet, dass es an der Einrichtung der Firewall-Regeln lag, weswegen eine komplette Neuinstallation vorgenommen wurde. Dies hat unnötig Zeit gefressen. Die Lösung war dann, die System zu Hause einzurichten.
- Durch den Zeitverzug, den das erste Problem mit sich trug, konnte die Dokumentation nicht rechtzeitig fertiggestellt werden.

Das aktualisierte Gantt-Diagramm ist in klein in Abb. 2.8 und groß im Anhang in Abb. A.2 auf Seite 19 zu sehen.

GANTT Diagramm Gruppe 7

PROJEKTTITEL	Projekt 3	COMPANY NAME	Projektteam IT20/2 Gruppe 7
PROJEKT MANAGER	Péter Egermann	DATE	11.12.2021

WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1	Projektplanung					100%
1.1	Pflichtenheft erstellen	Peter	15.11.2021	22.12.2021	28	100%
1.2	Projektstruktuplan erstellen	Vinzenz	15.11.2021	22.12.2021	28	100%
1.3	Netzplan überarbeiten	Gruppe 7	15.11.2021	22.12.2021	28	100%
2	Durchführung					100%
2.1	Server einrichten (DHCP, DNS)	Vinzenz	28.02.2022	08.03.2022	7	100%
2.2	Firewall einrichten	Vinzenz	03.01.2022	07.03.2022	16	100%
2.3	Netzwerk einrichten	Vinzenz	03.01.2022	01.03.2022	12	100%
2.4	Netzwerk testen	Vinzenz	06.01.2022	09.01.2022	2	100%
2.5	Firewall testen	Vinzenz	03.03.2022	10.03.2022	6	100%
2.6	Datenbankschema erstellen	Peter	28.02.2022	28.02.2022	1	100%
2.7	Datenbank einrichten	Peter	02.03.2022	03.03.2022	2	100%
2.8	Backendgrobplanung / PAP	Johannes	03.01.2022	05.01.2022	3	100%
2.9	Backend erstellen	Johannes	06.01.2022	11.01.2022	4	100%
2.10	Backend testen	Johannes	10.01.2022	12.01.2022	3	100%
2.11	Frontendgrobplanung / PAP	Johannes	03.01.2022	05.01.2022	3	100%
2.12	Frontend erstellen	Johannes	07.01.2022	12.01.2022	4	100%
2.13	Frontend testen	Johannes	11.01.2022	13.01.2022	3	100%
3	Projekt abschliessen					100%
3.1	Ablaufdokumentation	Peter	28.02.2022	14.03.2022	11	100%
3.2	Projektuebergabe	Peter	14.03.2022	14.03.2022	1	100%



Legende

UW3	KW46	15.11.2021	19.11.2021	
UW4	KW47	22.11.2021	26.11.2021	Meilenstein Phase 1
UW5	KW1	03.01.2022	07.01.2022	
UW6	KW2	10.01.2022	14.01.2022	Meilenstein Phase 2
UW7	KW9	14.03.2022	14.03.2022	Meilenstein Projektabgabe

Abbildung 2.8: Gantt-Diagramm

2.8 Optimierungsvorschläge zur Projektrealisierung

Das Projektteam IT20/2 Gruppe 7 bestand aus drei Anwendungsentwicklern, was die Sache deutlich erschwert hat. Hier wäre eine Überarbeitung der Gruppeneinteilung von Vorteil gewesen, so dass ein Systemintegrator und ein Anwendungsentwickler ein Team bilden, wodurch sich gewisse Synergieeffekte ergeben könnten.

Abbildungsverzeichnis

1.1 Schnittstellenspezifikation 4
1.2 Projektstrukturplan 5
1.3 Gantt-Diagramm 6

2.1 Firewall-Regeln 8
2.2 SSH-Login sowie Auflisten aller laufenden Python-Anwendungen 9
2.3 Stoppen einer bestimmten Python-Anwendungen 9
2.4 Starten einer bestimmten Python-Anwendungen 9
2.5 Überprüfen des DNS mittels Namens 10
2.6 Aufrufen des Frontends mittels Name 10
2.7 Einrichtung der Tickets-Datenbank 11
2.8 Gantt-Diagramm 13

Tabellenverzeichnis

1.1 Ansprechpartner Auftraggeber 1
1.2 Ansprechpartner Auftragnehmer 1

2.1 Netzwerkauslegung 7

Listings

2.1 Port-Freigabe einer CentOS-Firewall 10
2.2 Einrichtung Zugriff PostGreSQL 10
2.3 Ändern der tickets-Tabelle 12
2.4 Port-Freigabe einer CentOS-Firewall 12

Anhang

A Gantt-Diagramme

Abbildung A.1: Gantt-Diagramm

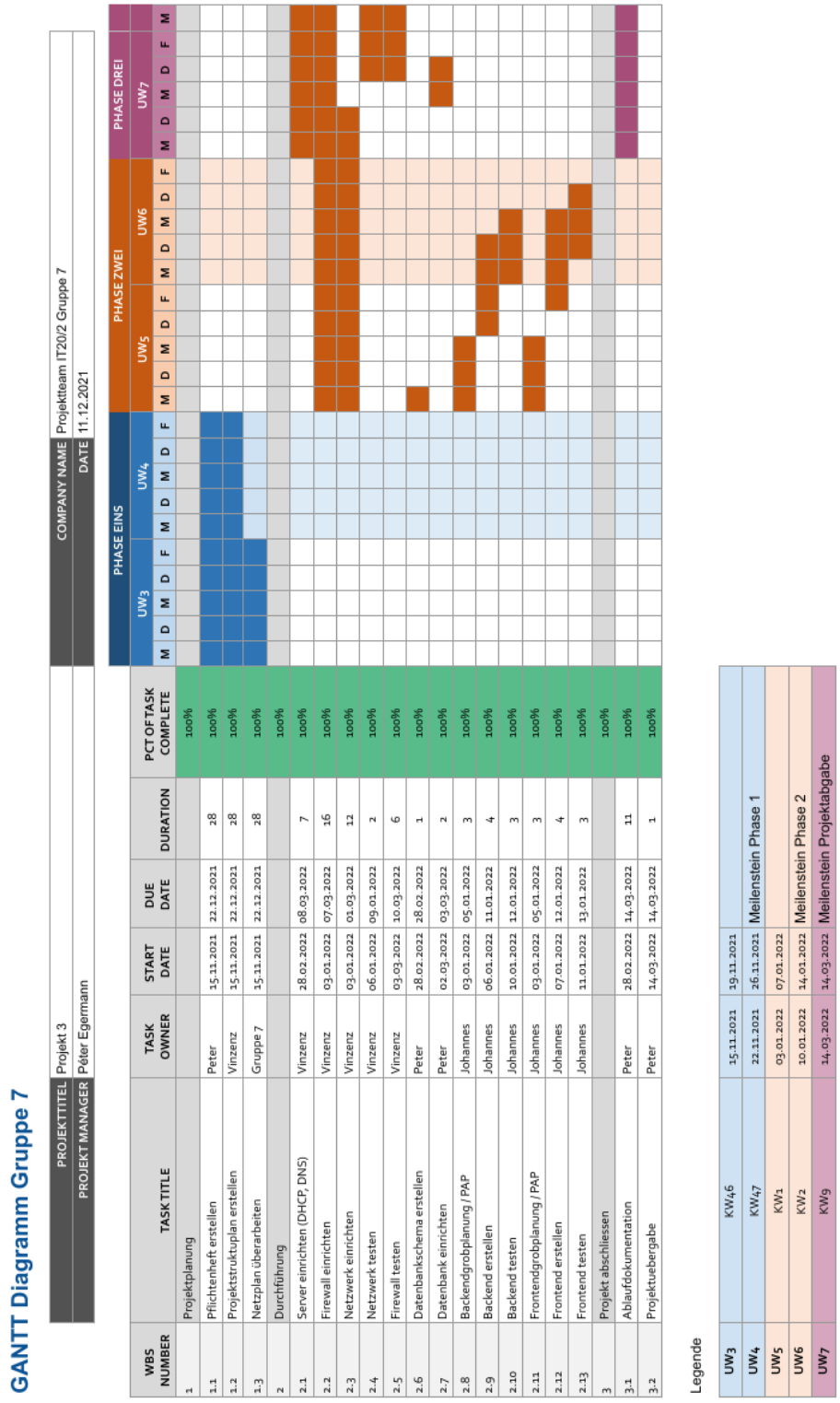


Abbildung A.2: Update Gantt-Diagramm

B Netzwerkplan

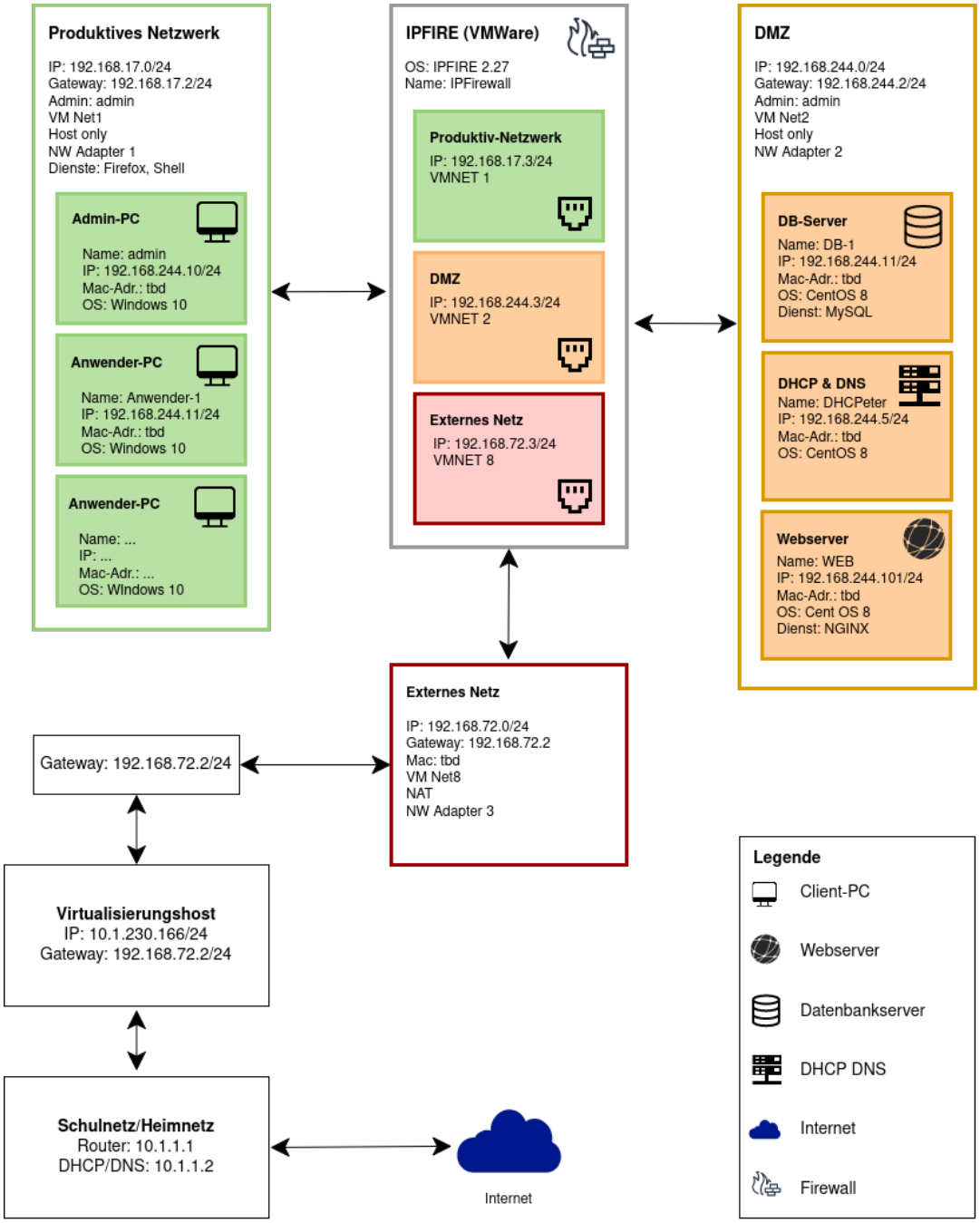


Abbildung B.1: Netzwerkplan