دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی

پیاده‌سازی سیستم پایش شبکه‌های کامپیوتری

نگارش

سیدمهدی میرفندرسکی

استاد راهنما

دکتر مسعود صبائی

شهریور ۱۴۰۱


بسم الله الرحمن الرحيم

به نام خدا

دانشگاه صنعتی امیرکبیر
(پلی‌تکنیک تهران)

تعهدنامه اصالت اثر

تاریخ: شهریور ۱۴۰۱

اینجانب سیدمهدی میرفندرسکی متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

سیدمهدی میرفندرسکی

امضا

سپاس‌گزاری

با سپاس فراوان از راهنمایی‌ها و زحمات استاد ارجمندم، جناب آقای دکتر مسعود صبائی، که از ابتدای راه و در طی انجام این پروژه، با رهنمودهایشان مرا در نگار این اثر یاری نمودند.

سیدمهدی میرفندرسکی

شهریور۱۴۰۱

چکیده

پیشرفت روز افزون شبکه‌های کامپیوتری باعث شده است تا مدیریت آن از اهمیت بالایی برخوردار باشد. به طور کلی مدیریت یک سیستم شامل پایش اجزا، تحلیل اطلاعات و انجام اقدامات برای نزدیک شدن به هدف آن سیستم خواهد بود. به بیانی دیگر مدیریت و به طور ویژه مدیریت شبکه‌های کامپیوتری یک فرایند دائمی شامل پایش، پردازش، برنامه‌ریزی و اقدام است. هدف از انجام این پروژه توسعه ابزاری جهت پایش شبکه‌های کامپیوتری بود. با توسعه این سامانه، اطلاعات مدیریتی شامل اطلاعات ترافیکی، اطلاعات پیکربندی، هشدارها و ... جمع‌آوری شده و به کاربر نمایش داده می‌شود. تا با این امکان، کاربر بتواند از طریق این اطلاعات، برنامه‌ریزی و اقدام کند. این سامانه از طریق یک واسط مدیریتی اطلاعات پایش را از اجزا با پروتکل SNMP دریافت می‌کند و به نحوی قابل فهم برای مدیر شبکه نمایش می‌دهد. اطلاعات مدیریتی پس از دریافت در سامانه، پردازش می‌شوند و در انتها در پایگاه‌های داده ذخیره خواهند شد. همچنین واسط کاربری تحت وبی نیز با واکشی این اطلاعات مدیریتی، آن‌ها را به کاربر نمایش می‌دهد. امکان دیگری که به این سامانه اضافه شد تا برنامه‌ریزی و اقدام را برای مدیریت تسهیل نماید، کشف شبکه بود. با این قابلیت کاربر می‌تواند یک دید کلی از شبکه نیز بدست آورد. درانتهای پروژه نیز نیازمندی‌های مختلف سامانه از جنبه کارکردی و غیرکارکردی آن مورد بررسی و تست قرار گرفتند.

واژه‌های کلیدی:

مدیریت شبکه، پایش شبکه، پروتکل SNMP، کشف شبکه

فهرست مطالب

عنوان صفحه

فهرست شکل‌ها

شکل صفحه

فهرست جدول‌ها

جدول صفحه

فصل اول

مقدمه

پیشرفت روز افزون شبکه‌های کامپیوتری باعث شده است تا مدیریت آن نیز از اهمیت بالایی برخوردار باشد. اگر بخواهیم مروری اجمالی بر مفهوم مدیریت داشته باشیم، این گونه بیان می‌کنیم که مدیریت یک سیستم شامل پایش[1] اجزا و جمع‌آوری اطلاعات، تحلیل اطلاعات و انجام اقدامات برای نزدیک شدن به هدف آن سیستم خواهد بود. به بیان دیگر مدیریت یک فرایند دائمی شامل پایش و نظارت، برنامه‌ریزی و اقدام است. مدیریت شبکه‌های کامپیوتری نیز شامل این فعالیت‌ها خواهد بود.

هدف از انجام این پروژه توسعه یک ابزار مدیریتی به منظور پایش و نظارت شبکه‌های کامپیوتری است. به عبارت دیگر در این پروژه، نرم‌افزاری توسعه داده می‌شود که به کمک آن امکان پایش و نظارت تجهیزات قابل مدیریت شبکه، فراهم گردد. این امکان از طریق یک واسط مدیریتی که اطلاعات پایش را از تجهیزات دریافت می‌کند و به نحوی قابل فهم برای مدیر شبکه نمایش می‌دهد، فراهم می‌گردد[۱].

با پیاده‌سازی این ابزار، مدیر شبکه می‌تواند مشکلات شبکه را به موقع متوجه شود. همچنین می‌تواند برنامه‌ریزی کوتاه‌مدت و بلندمدت به منظور استفاده بهینه از منابع و جلوگیری از خرابی، انجام دهد. پس از نگاه دقیق تری به مسئله پایش شبکه، حال ویژگی‌های مختلفی که این سامانه باید داشته باشد در ادامه بیان می‌شود:

- انتخاب پروتکل مناسب و نحوه ارتباط: دریافت اطلاعات مدیریتی از عناصر باید از طریق یک پروتکل صورت بگیرد که به این نوع پروتکل‌ها، پروتکل مدیریتی گفته می‌شود. اتخاذ یک پروتکل مدیریتی ایمن که مصرف پهنای باند حداقلی را داشته باشد بسیار مهم است. این پروتکل‌ها می‌توانند مانندSNMP[2]، استاندارد شده باشند و یا حتی می‌توانند مانند CLI[3] غیر استاندارد باشند. اکثر عناصر تحت مدیریتی از پروتکل‌های SNMP و CLI پشتیبانی می‌کنند. همچنین دستگاه‌های ویندوزی از پروتکل WMI[4] نیز پشتیبانی می‌کنند. پروتکل SNMP یکی از پروتکل‌های لایه کاربرد[5] برای مدیریت و پایش عملکرد عناصر شبکه است. درواقع برای ارتباط با سیستم مدیریت شبکه تنها باید پیکربندی و فعال شود. به بیان دیگر نیازی به توسعه برنامه‌ای در سمت عناصر شبکه وجود ندارد[۲]. این پروژه قصد استفاده از پروتکل SNMP را دارد.

[1] Monitoring
[2] Simple Network Management Protocol
[3] Command-line interface
[4] Windows Management Instrumentation
[5] Application layer

- کشف شبکه[1] و دستگاه‌های مورد نیاز برای پایش: در پایش شبکه، اولین قدم شناسایی عناصر قابل مدیریت و معیارهای عملکرد مرتبط با هر عنصر است. عناصری مانند رایانه‌های رومیزی و چاپگرها و مواردی از این دست برای ما حائز اهمیت نیستند و اساسا نیازی به پایش مداوم ندارند. از طرفی سرورها[2]، مسیریاب‌ها[3] و سوئیچ‌ها[4] وظایفی حیاتی را بر عهده دارند و نیاز به پایش مداوم دارند. ابزار پایش شبکه باید قادر باشد تا عناصر تحت مدیریت را شناسایی کند.
- دریافت اطلاعات و تنظیم پارامترهای مختلف: دریافت اطلاعات از عناصر شبکه که قابل مدیریت باشند، به دو صورت انجام می‌پذیرد:

۱. ارسال درخواست سیستم مدیریت به عنصر تحت مدیریت و دریافت پاسخ از سمت عنصر تحت مدیریت

۲. ارسال اعلان[5] بر اساس رخداد و یا رفتار غیر متعارف توسط عنصر تحت مدیریت و دریافت در سمت سیستم مدیریت

روش[6] توسعه این سامانه بدین صورت است که، ابتدا نیازمندی‌ها جمع‌آوری می‌شوند، بعد از آن فناوری‌های[7] توسعه سامانه انتخاب می‌شوند. حال که فناوری سامانه تعیین شد، معماری سامانه طراحی می‌شود. بعد از طراحی معماری، پیاده‌سازی سامانه صورت می‌گیرد. درنهایت نیز سامانه بر اساس تحلیل نیازمندی‌ها تست می‌شود.

همچنین مراحل گفته شده برای توسعه این سامانه در ادامه این پایان‌نامه آورده شده است. ابتدا فصل دوم مروری بر سامانه‌های مشابه خواهد داشت. سپس فصل سوم به ترتیب به جمع‌آوری نیازمندی‌ها و تحلیل آن‌ها، انتخاب فناوری و طراحی معماری می‌پردازد. در فصل چهارم شیوه پیاده‌سازی این سامانه بیان می‌شود و در فصل پنجم تست‌های صورت گرفته مطرح خواهند شد. در فصل آخر نیز از نتیجه‌گیری و کارهای آینده مطالبی بیان خواهد شد.

[1] Network Discovery
[2] Servers
[3] Routers
[4] Switches
[5] Notification
[6] Methodology
[7] Technology

فصل دوم

مروری بر سامانه‌های مشابه

از زمانی که دستگاه‌ها در شبکه‌ها به هم متصل می‌شدند، نیاز به نوعی سیستم مدیریت و پایش شبکه وجود داشته است. در سال ۱۹۸۸ م. بود که SNMP به استاندارد جدید تبدیل شد. هدف پروتکل SNMP این است که زبانی برای انتقال اطلاعات مدیریتی شبکه بین دستگاه‌های مختلف به وجود آورد. امروزه اکثر دستگاه‌های شبکه می‌توانند SNMP را به عنوان یک عامل راه‌اندازی کنند، به همین جهت اکثر نرم‌افزارهای پایش شبکه ارتباط از طریق SNMP را در اولویت خود قرار می‌دهند[۳]. از سال‌ها پیش با طراحی اینگونه سامانه‌ها کارهای ارزشمندی انجام شده است. این فصل به معرفی اجمالی بعضی از این سامانه‌ها می‌پردازد.

۲-۱ مروری بر سامانه‌های موجود

امروزه ابزارهای متنوع و گوناگونی برای پایش شبکه توسعه داده شده‌اند. در ادامه ابزارهای سولاروبندز[۱]، دیتاداگ[۲] و زبیکس[۳] معرفی می‌شوند.

۲-۱-۱ سولاروبندز

این سیستم پایش شبکه از SNMP برای بررسی وضعیت عناصر تحت مدیریت استفاده می‌کند. این ابزار قابلیت کشف عناصر شبکه را داراست. به عبارتی دیگر دستگاه‌های موجود در شبکه که برای ما حائز اهمیت هستند را پیدا کرده و توپولوژی شبکه را ترسیم می‌کند. همچنین می‌توان یک توپولوژی مناسب برای کل زیرساخت شبکه طراحی کرد. به علاوه هشدارهای هوشمندی نیز دارد[۴]. اما در بحث نصب بر روی سیستم عامل‌های مختلف محدودیت‌هایی وجود دارد، مثلا در بعضی توزیع‌های لینوکس[۴] بر پایه دبین[۵] قابل نصب نیست.

[1] SolarWinds
[2] Datadog
[3] Zabbix
[4] Linux
[5] Debian

برخی دیگر از ویژگی‌های سولاروینداز عبارتند از:

- قابلیت تجزیه و تحلیل مشکل: با فراهم آوردن دید کاملی از عملکرد زیرساخت شبکه، به هنگام وجود آمدن مشکل، پیدا کردن مبدا آن ساده خواهد بود.
- پایش عملکرد: این امکان را می‌دهد که بتوان بررسی کرد آیا اهداف عملکردی سرویس‌های مختلف برآورده شده‌اند یا خیر. این با پایش عملکرد در سطح برنامه‌های کاربردی، محقق می‌شود.
- سهولت استفاده: واسط کاربری کاربرپسند و ساده‌ای دارد.

همچنین از معایب این سامانه می‌توان به موارد زیر اشاره کرد:

- در صورت عدم سفارشی‌سازی هشدارهای دریافتی، مقدار زیادی هشدار دریافت می‌شود و چون مقدار آن‌ها زیاد است، توسط کاربر نادیده گرفته می‌شوند.
- برای بعضی کاربران این سامانه، واسط کاربری گاهی اوقات می‌تواند گیج‌کننده باشد.
- برای بعضی کاربردها هشدارها مبهم می‌باشد و نیاز کاربران برطرف نمی‌شود.
- برای استفاده از این سامانه به طور کامل باید هزینه پرداخت شود.

۲-۱-۲ دیتاداگ

دیتاداگ یک ابزار پایش عملکرد شبکه است که مبتنی بر ابر[1] است و این امکان را می‌دهد که ترافیک شبکه بین میزبان‌ها[2]، کانتینرها[3] و سرویس‌ها را در ابر تحلیل کنیم[۵].

[1] Software as a service
[2] Hosts
[3] Containers

برخی امکانات و نقاط قوتی که دیتاداگ دارد به شرح زیر می‌باشد:

- این برنامه جریان ترافیک شبکه را می‌تواند بین میزبان‌ها، کانتینرها، شبکه‌های مختلف و حتی مفاهیم انتزاعی مانند سرویس‌ها و یا ماژول‌های مختلف نمایش می‌دهد.
- داده‌های ترافیک شبکه را با درنظر گرفتن برنامه‌های مربوطه، معیارهای دستگاه‌های مختلف و لاگ‌ها تحلیل می‌کند تا عیب‌یابی را در یک سیستم انجام دهد.
- به صورت بصری جریان ترافیک را نشان می‌دهد تا به شناسایی گلوگاه‌های ترافیکی کمک کند.

همچنین از معایب این سامانه می‌توان به موارد زیر اشاره کرد:

- استفاده از این سامانه برای کاربران جدید شاید بسیار سخت باشد ازطرفی مستندسازی شیوا و فصیحی ندارد، به همین دلیل باید زمانی صرف ارتباط با پشتیبانی شود.
- همچنین گزارش شده است که گاهی اوقات درک نمودارها بسیار دشوار است و استفاده از آن به دانش فنی نیاز دارد.
- همچنین مانند سولارویندز برای استفاده از آن، باید هزینه پرداخت شود.

۲-۱-۳ زبیکس

زبیکس یک ابزار پایش شبکه متن‌باز[1] است که برای انواع عناصر تحت مدیریت خدمات ارائه می‌دهد. برخی امکاناتی که زبیکس در اختیار ما قرار می‌دهد به شرح زیر است[۶]:

- جمع‌آوری اطلاعات انعطاف‌پذیر است و با تغییر شبکه مشکلی پیش نخواهد آمد.
- توانایی شناسایی دستگاه‌های شبکه به طور خودکار را داراست.
- امکانات مختلفی برای هشدارها ارائه می‌دهد.

[1] Open-Source

همچنین از معایب این سامانه می‌توان به موارد زیر اشاره کرد:

- برخی از خطاها اطلاعات کافی برای عیب‌یابی ارائه نمی‌دهند.
- برخی مستندات این سامانه باید بروز شوند.
- انعطاف‌پذیری دیرهنگام با سرویس‌های ابری مانند AWS[1]

۲-۲ خلاصه

در این فصل بر روی سامانه‌های مشابه در حوزه پایش شبکه، مروری انجام شد. ابتدا تاریخچه و اهمیت پروتکل SNMP ارائه شد. بعد از آن به ترتیب به سه ابزار سولارویندز، دیتاداگ و زبیکس به صورت خلاصه پرداخته شد. از نقاط ضعف سامانه سولارویندز به قابل نصب نبودن بر روی برخی توزیع‌های لینوکس بر پایه دبین، هزینه‌بر بودن آن و تعداد هشدار بالای آن اشاره شد. البته نقاط قوت خوبی ازجمله قابلیت تجزیه و تحلیل مشکل و سهولت استفاده نیز ذکر شد. بعد از مطرح کردن سولارویندز، دیتاداگ که یک ابزار پایش مبتنی بر ابر بود معرفی شد. از نقاط ضعف این سامانه به کیفیت پایین مستندات آن، هزینه‌بر بودن آن و نیاز به دانش فنی برای درک نمودارهای آن اشاره شد. همچنین برای نقاط قوت به نشان دادن جریان ترافیک به صورت بصری بین میزبان‌ها، کانتینرها، شبکه‌های مختلف و سرویس‌ها اشاره شد. درنهایت نیز به سامانه زبیکس که نقطه قوت متمایز آن رایگان و متن‌باز بودن آن بود، پرداخته شد. البته معایبی مثل عدم ارائه اطلاعات کافی خطاها برای آن نیز ذکر شد.

[1] Amazon Web Services

فصل سوم

تحلیل و طراحی

هدف از این فصل بررسی نیازمندی‌ها، تحلیل و طراحی معماری سامانه است. سامانه ذکر شده باید علاوه بر داشتن ویژگی‌های یک سیستم پایش شبکه، باید استفاده آن برای کاربر راحت و با کمترین دانش فنی قابل استفاده باشد. برای این که بتوان چنین سامانه‌ای را طراحی کرد ابتدا باید نیازمندی‌های سامانه را تشخیص داد و سپس معماری کلی سامانه موردنظر را بر اساس فناوری‌های انتخابی به دست آورد.
در این فصل ابتدا نیازمندی‌ها تحلیل می‌شوند. بعد از آن فناوری‌های توسعه نرم‌افزار بررسی و انتخاب می‌شوند. و درنهایت معماری نرم‌افزار ترسیم می‌شود.

۳-۱ تحلیل نیازمندی‌ها

بر اساس مطالعاتی که انجام شده و مطالب فصل قبل این سامانه باید ویژگی‌های زیر را داشته باشد:

- کشف شبکه و ترسیم بصری آن برای ادراک بهتر توسط مدیر شبکه
- توانایی تنظیم پارامترهای مختلف عملکردی توسط مدیر شبکه از جمله دوره تناوب جمع‌آوری اطلاعات از عناصر
- جمع‌آوری اطلاعات مدیریتی و پردازش آن‌ها جهت تولید اطلاعات قابل فهم توسط مدیر شبکه
- ایجاد یک واسط کاربری مطلوب تحت وب و نمایش اطلاعات قابل فهم
- فراهم آوردن حداقلی امنیت سیستم با استفاده از پروتکل‌های ایمن
- مقیاس پذیری سیستم جهت کارآمد بودن در هنگام افزایش وسعت شبکه
- فراهم آوردن یک بستری برای دریافت هشدارهای ارسالی عناصر تحت مدیریت

به طور خلاصه هدف از انجام این پروژه توسعه یک ابزار پایش شبکه با ویژگی‌های پایه‌ای فوق است. به عبارتی هدف، افزودن یک امکان جدید به پروژه‌های موجود و پیاده‌سازی آن نیست، اما از زبان‌های برنامه‌نویسی و فناوری‌های بروز در مقایسه با پروژه متن‌باز زبیکس استفاده خواهد شد. به عبارت دیگر این پروژه از ابتدا بدون استفاده از کدهای متن‌باز موجود توسعه داده خواهد شد.

در ادامه نیز بعد از تحلیل نیازمندی‌های پروژه نیازمندی‌های کارکردی[1] و غیرکارکردی[2] در دو بخش مجزا توضیح داده می‌شوند.

۳-۱-۱ نیازمندی‌های کارکردی

نیازمندی‌های کارکردی، کارکردها و وظایف یک سیستم و اجزای آن را مشخص می‌کند، کارکرد به عنوان مجموعه‌ای از ورودی‌ها، رفتار و خروجی‌ها تعریف می‌شود؛ در واقع نیازمندی‌های کارکردی وظایفی است که یک سیستم موظف به انجام آن می‌باشد[۷]. در جدول ۳-۱ به شرح نیازمندی‌های کارکردی پرداخته می‌شود.

۳-۱-۲ نیازمندی‌های غیرکارکردی

نیازمندی‌های غیرکارکردی به ویژگی‌های کیفی، محدودیت و قیود یک سیستم اطلاق می‌شود که در توسعه معماری و طراحی سیستم باید مدنظر قرار گیرند[۷]. در این سیستم نیازمندی‌های غیرکارکردی شامل کارایی، سهولت ستفاده و امنیت خواهد بود که در جدول ۳-۲ به توضیح آن پرداخته شده است.

[1] functional requirements

[2] non functional requirements

جدول ۳-۱: جدول نیازمندی‌های کارکردی

ردیف	عنوان	شرح
۱	کشف شبکه	سامانه باید قادر باشد تا با دریافت یک آدرس شبکه، عناصری که عامل SNMP بر روی آن‌ها فعال هستند را به همراه نوع عنصر (سرور، مسیریاب، سوئیچ و تکرارکننده) آن‌ها مشخص کند. این خروجی باید در قالب یک توپولوژی شبکه باشد.
۲	تنظیمات شبکه	سامانه باید قادر باشد مشخصات عناصر مختلف از جمله نام کاربری، رمز عبور و ... را از کاربر دریافت کرده و مراحل بعدی طبق این مشخصات طی شوند. این مشخصات شامل پارامترهایی که کاربر جهت پایش مشخص می‌کند نیز می‌باشد.
۳	پردازش و ذخیره اطلاعات جمع‌آوری شده	این سامانه باید بتواند اطلاعاتی که از شبکه جمع‌آوری می‌کند را پردازش و همچنین آن‌ها را در خود ذخیره کند.
۴	جمع‌آوری اطلاعات شبکه بر اساس تنظیمات شبکه	این سامانه باید بتواند بر اساس تنظیماتی که کاربر بر شبکه اعمال کرده است، وضعیت هر عنصر تحت مدیریت را رصد و پایش کند.
۵	جمع‌آوری هشدارهای مربوط به عناصر تحت مدیریت	این سامانه باید بتواند هشدارهایی که از سمت شبکه به سامانه وارد می‌شوند را به کاربر نمایش دهد.

جدول ۳-۲: جدول نیازمندی‌های غیر کارکردی

ردیف	عنوان	کیفیت
۱	مدت زمان کشف شبکه	شرح این کارکردی نیازمندی در جدول ۳-۱ آمده است. این نیازمندی برای هر شبکه باید کمتر از دو دقیقه زمان ببرد.
۲	واسط کاربری مطلوب تحت وب	این سامانه باید بتواند واسط کاربری تحت وبی ارائه کند، تا از راه دور در دسترس باشد. همچنین این واسط کاربری باید زیبا و کار با آن آسان باشد.
۳	امنیت سامانه	این سامانه باید جنبه‌های مختلف امنیت را در نظر گرفته و ایمنی حداقلی را فراهم آورد.

۳-۲ بررسی و انتخاب فناوری پیاده‌سازی

برای طراحی معماری سامانه ابتدا نیاز است تا فناوری‌های مورد استفاده معرفی شوند. به طور کلی نیز با توجه به نیازمندی‌ها، فناوری‌ها در چهار گروه واسط کاربری، سمت سرور[۱]، هسته SNMP و ذخیره‌سازی اطلاعات تقسیم می‌شوند. برای هر گروه در این بخش، ابتدا فناوری‌های موجود بررسی و درنهایت فناوری مورد استفاده نیز مشخص می‌شود.

۳-۲-۱ واسط کاربری

برای توسعه واسط کاربری، هزینه توسعه پایین، زیبا بودن و راحتی کار با آن باید در نظر گرفته شود. البته ذکر این نکته نیز لازم است که واسط کاربری باید تحت وب باشد تا از راه دور نیز قابل دسترس باشد. برای توسعه واسط کاربری با این مشخصات، چارچوب‌های[۲] ری‌اکت[۳]، انگولار[۴] و ویو جی‌اس[۵] مطرح هستند.

[1] Back-end
[2] Frameworks
[3] React
[4] Angular
[5] Vue.js

۳-۲-۱-۱ ری‌اکت

یک کتابخانه متن‌باز توسعه یافته توسط فیسبوک است. این کتابخانه طبق نظرسنجی توسعه‌دهندگان استک اورفلو[1] در سال ۲۰۲۱ م. توسط اکثر توسعه‌دهندگان واسط کاربری استفاده شده است[۸]. هدف اصلی این چارچوب رفع مشکلات نگهداری کد به دلیل استفاده مداوم کدها در برنامه است.

ری‌اکت به دلیل مدل شیء سند مجازی[2]، عملکردی عالی از خود ارائه می‌دهد. همچنین برای کاربردهای با ترافیک بالا مناسب است. از طرفی به دلیل وجود مستندات کافی، برای توسعه‌دهندگان جدید یادگیری آن نسبتا راحت خواهد بود[۹].

مزایا:

- صرفه‌جویی در زمان در هنگام استفاده مجدد از اجزا
- یک چارچوب متن‌باز با ابزارهای متنوع
- مناسب برای برنامه‌های تک صفحه‌ای
- جامعه توسعه‌دهندگان قابل توجه

معایب:

- زمان یادگیری نسبتا طولانی
- چالش برانگیز بودن درک پیچیدگی‌های JSX[3] برای توسعه‌دهندگان

[1] Stack Overflow Developer Survey
[2] Virtual Document Object Model (Virtual DOM)
[3] JavaScript XML

۳-۲-۱-۲ انگولار

این چارچوب که بر اساس تایپ‌اسکریپت[1] است، به طور رسمی در سال ۲۰۱۶ م. منتشر شد. انگولار توسط گوگل ایجاد شد تا شکاف بین نیازهای روزافزون فناوری و مفاهیم را کاهش دهد. برخلاف ری‌اکت، انگولار ویژگی اتصال داده دو طرفه[2] را داراست. این بدین معنی است که یک همگام‌سازی زمانی بین نمایش[3] و مدل وجود دارد. به عبارت دیگر هر تغییری در مدل به سرعت در نمایش و برعکس تکرار می‌شود. در مقایسه انگولار در مقابل ری‌اکت، یادگیری انگولار آسان نخواهد بود. با این حال، مستندات بی‌شماری در دسترس است[۱۰].

مزایا:

- فرایند آسان تولید کد
- جامعه توسعه‌دهندگان قابل توجه
- عملکرد بالا
- سازگار با معماری‌های MVC[4] و MVVM[5]

معایب:

- پیچیده بودن انگولار
- دشوار بودن جابجا کردن طرح‌های قدیمی از انگولار بر اساس جاوا اسکریپت به انگولار بر اساس تایپ‌اسکریپت
- زیاد بودن تلاش یادگیری

[1] Typescript
[2] Two-way Data Binding
[3] View
[4] Model–View–Controller
[5] Model-View-ViewModel

۳-۲-۱-۳ ویو جی‌اس

یکی از ساده‌ترین چارچوب‌ها ویو جی‌اس است. با وجود اندازه کوچک آن دو مزیت اصلی دارد:

- وجود مدل شی سند بصری
- مبتنی بر مؤلفه بودن آن

همچنین از اتصال داده دو طرفه مانند انگولار بهره می‌برد.

تفاوت ویو جی‌اس و ری‌اکت در این است که ویو جی‌اس یک چارچوب جاوا اسکریپت[1] است در حالی که ری‌اکت یک کتابخانه جاوا اسکریپت است. بنابراین ویو جی‌اس برای پروژه‌های بزرگ مناسب‌تر است. اگرچه ویو جی‌اس برای مقابله با پیچیدگی‌ها و بهبود عملکرد برنامه ایجاد شده است، اما هنوز در میان صنعت‌های بزرگ محبوبیت زیادی ندارد. به طور مشابه، با مقایسه انگولار در مقابل ویو جی‌اس، ویو جی‌اس عملکرد انگولار را بهبود می‌بخشد.

مزایا:

- مستندات کامل و دقیق
- سادگی و وضوح
- دارای ابزارهای توسعه‌دهنده مرورگر
- قابلیت استفاده مجدد کد و یکپارچه سازی ساده

معایب:

- کوچک بودن جامعه توسعه‌دهندگان
- بی نظمی در کد به دلیل انعطاف‌پذیری

۳-۲-۱-۴ جمع‌بندی

با توجه به موارد ذکر شده و زمان محدود برای پیاده‌سازی پروژه، کتابخانه ری‌اکت برای توسعه واسط کاربری استفاده شد. از جمله دلایل این انتخاب می‌توان به کوچک بودن جامعه توسعه‌دهندگان ویو جی‌اس و پیچیده بودن انگولار اشاره کرد. از طرفی نیازمندی‌های پروژه از امکاناتی که کتابخانه ری‌اکت در اختیار ما قرار می‌دهد، فراتر نخواهد رفت.

[1] Javascript

۳-۲-۲ سمت سرور

برای توسعه سمت سرور و نوشتن APIs بهترین گزینه‌های موجود فلسک[۱]، جنگو[۲] و نودجی‌اس[۳] بودند که در ادامه معرفی خواهند شد.

۳-۲-۲-۱ فلسک

فلسک یک چارچوب وب است که به زبان پایتون[۴] نوشته شده است. ساختار فلسک واضح‌تر از چارچوب جنگو است و همچنین یادگیری آن آسان‌تر است زیرا کد کمتری برای پیاده‌سازی یک وب اپلیکیشن ساده نیاز دارد. فلسک از انعطاف بالایی برخوردار است. و همچنین از عملکردی ثابت در سراسر اجرای برنامه برخوردار است.

۳-۲-۲-۲ جنگو

جنگو نیز مانند فلسک یک چارچوب پایتون است که ساخت وب سایت با استفاده از پایتون را آسان‌تر می‌کند. جنگو از الگوی طراحی MVT[۵] به صورت زیر پیروی می‌کند[۱۱]:

- مدل: ارائه داده‌های مطلوب از پایگاه داده را برعهده دارد.
- نمایش: یک کنترل‌کننده برای درخواست ورودی است که الگو و محتوای مربوطه را بر اساس درخواست کاربر برمی‌گرداند.
- الگو: یک فایل متنی حاوی طرح‌بندی صفحه وب، با منطق نحوه نمایش داده‌ها است.

[1] Flask
[2] Django
[3] Node.js
[4] Python
[5] Model View Template

۳-۲-۲-۳ نودجی‌اس

نودجی‌اس موتور جاوا اسکریپت وی هشت[1]، هسته گوگل کروم را خارج از مرورگر اجرا می کند.

یک برنامه نودجی‌اس در یک فرآیند[2] واحد (بدون ایجاد یک نخ[3] جدید برای هر درخواست) اجرا می‌شود. نودجی‌اس مجموعه‌ای از ورودی/خروجی‌های ابتدایی ناهمزمان را در کتابخانه استاندارد خود ارائه می‌دهد که از بلاک شدن کد جاوا اسکریپت جلوگیری می‌کند. این به نودجی‌اس اجازه می‌دهد تا هزاران اتصال همزمان را با یک سرور بدون وارد کردن بار مدیریت همزمانی نخ‌ها، که می‌تواند منبع مهمی از خرابی باشد، مدیریت کند[۱۲].

۴-۲-۲-۳ جمع‌بندی

در قسمت توسعه سمت سرور فلسک انتخاب شد که دلایل آن را ادامه می‌بینید:

- جنگو یک چارچوب برای توسعه همه‌جانبه[4] یک سایت است که به دلیل آن که در این سامانه می‌خواهیم از APIs استفاده کنیم جنگو مناسب این پروژه نخواهد بود.
- فلسک در مقایسه با نودجی‌اس کارایی کمتر و حتی سرعت پایین‌تری دارد، اما توجه به این نکته لازم است که سرعت این سامانه برای مدیریت درخواست‌های زیاد کاربردی برای ما نخواهد داشت؛ زیرا کاربران این سامانه محدود هستند.
- از طرفی به دلیل تولید داده بسیار توسط این سامانه، پایتون به دلیل وجود کتابخانه‌های متنوع و کارامد گزینه مناسب‌تری خواهد بود.

[1] V8 JavaScript engine
[2] Process
[3] Thread
[4] Full Stack

۳-۲-۳ هسته SNMP

برای توسعه ابزاری برای مدیریت پیام‌های SNMP به چندین زبان و کتابخانه می‌توان این کار را انجام داد. در ادامه برای سه زبان نکاتی که وجود دارد بیان می‌شود:

- زبان C: انتخاب به نسبت مناسبی خواهد بود. از این جهت که حتی ابزارهای نوشته شده برای لینوکس مانند snmpget و ... با کتابخانه net-snmp نوشته شده‌اند.
- زبان پایتون: برای توسعه این قسمت با زبان پایتون نیاز به استفاده از ماژول PySNMP وجود دارد که به شدت کند است و با تعداد کمی از عناصر تحت مدیریت بر پردازنده سرور فشار زیادی وارد می‌کند.
- زبان ارلنگ[1]: زبان ارلنگ به صورت داخلی از SNMP پشتیبانی می‌کند. و در آزمایشاتی حتی از زبان‌های C یا C++ بهتر عمل کرده است[۱۳]. اما با توجه به محدودیت زمانی توسعه پروژه و همچنین نا آشنا بودن با زبان ارلنگ این زبان بررسی نشد.

و درنهایت از زبان C و کتابخانه net-snmp برای توسعه انتخاب شد.

۴-۲-۳ ذخیره‌سازی اطلاعات

در پیاده‌سازی این سامانه اطلاعاتی که باید ذخیره شوند به شرح زیر می‌باشد:

- هشدارهای دریافتی از سمت عناصر تحت مدیریت
- اطلاعات جمع‌آوری شده برای برای یک پارامتر خاص از عناصر تحت مدیریت
- اطلاعات کاربرانی که توسط یک مدیر ارشد تعریف می‌شوند
- اطلاعات شبکه (توپولوژی، عناصر تحت مدیریت و ...)

[1] Erlang

داده‌های مربوط به دو مورد اول به صورت خیلی سریع تولید می‌شوند. همچنین نیاز به بازیابی سریع و مداوم آن‌ها نیز موجود دارد. اما از طرفی داده‌های مربوط به دو مورد آخر هم دارای ساختار هستند و هم مقدار دسترسی به این نوع داده‌ها زیاد نیست. با توجه به موارد ذکر شده برای دو مورد اول به دلیل داده زیاد و سرعت بازیابی بهتر از پایگاه‌داده‌های در حافظه اصلی[1] استفاده می‌کنیم. همچنین برای دو مورد بعدی از پایگاه‌داده‌های رابطه‌ای[2] مناسب هستند.

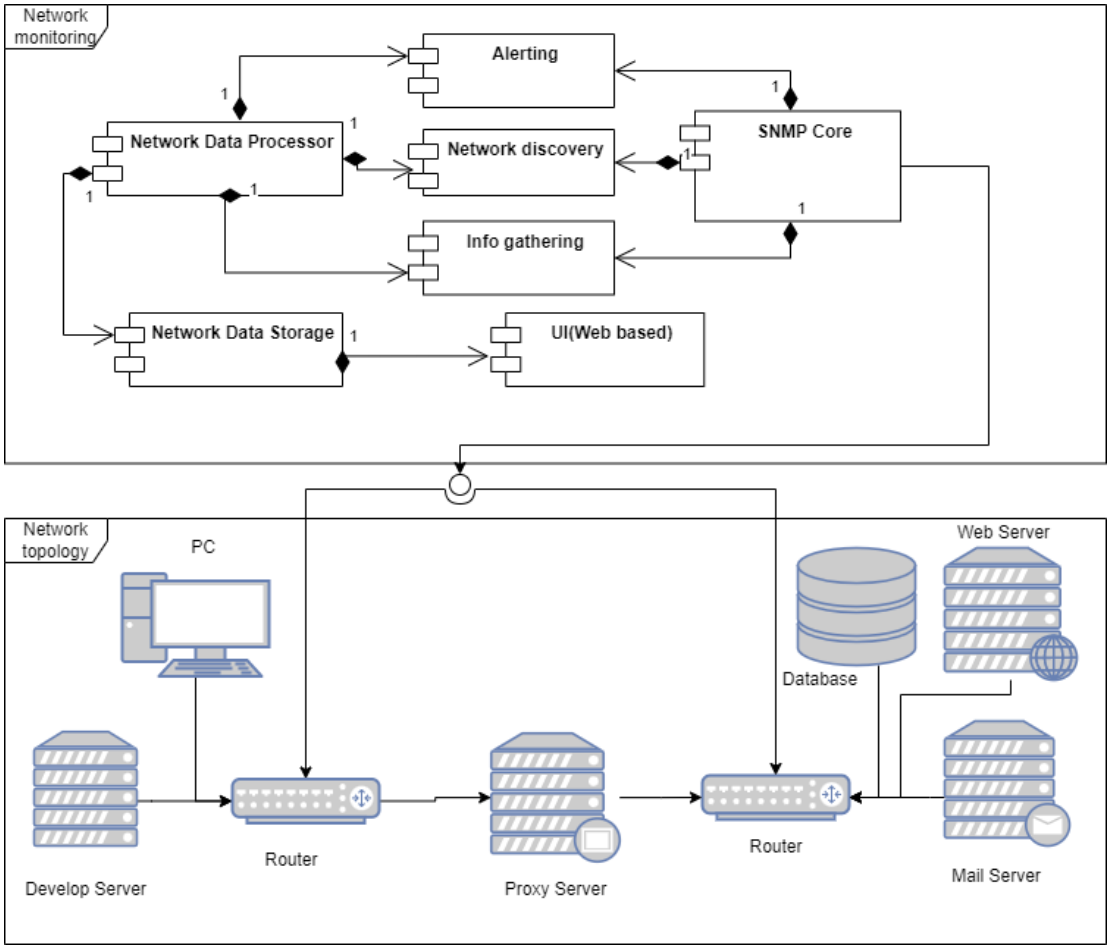
برای انتخاب پایگاه‌داده در حافظه اصلی یکی از بهترین گزینه‌ها ردیس[3] خواهد بود. ردیس به دلیل رایگان بودن، سرعت بازیابی بالا، پایین بودن حجم داده ذخیره شده و ... نسبت به رقبا برتری محسوسی دارد.

برای انتخاب پایگاه‌داده رابطه‌ای نیز با توجه به چارچوب انتخابی سمت سرور (فلسک) SQLite بهترین انتخاب خواهد بود. به طور کلی از سازگارترین پایگاه‌های داده با فلسک میتوان به SQLite و MySQL اشاره کرد. از طرفی برای کار با SQLite نیاز به نصب هیچ بسته نرم‌افزاری برای کار با آن نیست و در پایتون تعبیه شده است.

[1] In-Memory Database
[2] Relational Databases
[3] Redis

۳-۳ معماری سامانه

با توجه به مطالبی که در این فصل تا به اینجا گفته شد، سامانه پایش شبکه‌های کامپیوتری به ماژول‌های زیر تقسیم می‌شود شکل ۳-۱. ابتدا یک ماژول تحت عنوان هسته SNMP در نظر گرفته می‌شود که وظیفه مدیریت پیام‌ها و پیاده‌سازی پروتکل به یک زبان برنامه‌نویسی خاص است.



شکل ۳-۱: نمودار بلوکی اجزای سامانه

سپس ماژول کشف عناصر تحت مدیریت شبکه در نظر گرفته می‌شود، که به کمک ماژول هسته وظیفه جمع‌آوری اطلاعات ساختاری شبکه را بر عهده دارد. بعد از آن ماژول جمع‌آوری اطلاعات، عملکرد کل شبکه را رصد می‌کند. حال اگر زمانی با توجه به وجود اعلان‌ها نیاز به هشدار وجود داشت، از ماژول هشدار استفاده می‌شود.

نکته حائز اهمیت در رابطه با سه ماژول آخر این است که هر ماژول از ماژول هسته استفاده می‌کند. همچنین اطلاعاتی که هر ماژول بدست می‌آورد تحویل ماژول پردازش اطلاعات می‌دهد.

در پردازشگر اطلاعات شبکه، اطلاعات خام دریافتی از سه ماژول هشدار، کشف شبکه و جمع‌آوری اطلاعات پردازش می‌شوند تا اطلاعات قابل فهم توسط مدیر استخراج شود. حال باید اطلاعات تولید شده به ماژول ذخیره‌سازی اطلاعات داده شود.

ماژول واسط کاربری نیز در قالب یک وب سایت و فراهم آوردن یک پنل ورودی برای مدیران شبکه نیز اطلاعات ساختاری شبکه، پایش شبکه و هشدارها را از ماژول ذخیره‌سازی دریافت کرده و نمایش می‌دهد. همچنین از طریق آن می‌توان پارامترهای مختلف برای عناصر مختلف تنظیم و اقدام به اسکن کل شبکه کرد.

اما نیاز است که یک واسطی بین شبکه و سامانه مذکور باشد. در شکلی که بررسی شد، سامانه به یک شبکه فرضی از طریق مسیریاب‌های آن متصل است. در واقع واسط بین سامانه و شبکه ماژول هسته SNMP خواهد بود.

۳-۴ خلاصه

هدف نهایی این فصل طراحی معماری سامانه بود. برای این امر ابتدا زیرمجموعه‌ای از ویژگی‌های سامانه‌های فصل دوم ذکر شد. سپس با تحلیل نیازمندی‌ها دو جدول نیازمندی‌های کارکردی و غیرکارکردی تولید شد. در نهایت بعد از تولید نیازمندی‌ها به بررسی فناوری‌های پیاده‌سازی بخش‌های مختلف پرداخته شد. در نهایت نیز با توجه به فناوری‌های انتخاب شده، معماری جهت توسعه نرم‌افزار طراحی شد.

فصل چهارم

پیاده‌سازی

در بخش قبلی معماری کلی سامانه و ماژول‌های موردنیاز برای پیاده‌سازی سامانه پایش شبکه بیان و به صورت اجمالی معرفی شدند. این فصل به چگونگی قرار گرفتن و ارتباط بخش‌های مختلف می‌پردازد و پیاده‌سازی سیستم را توضیح خواهد داد.

ماژول‌های تعریف شده در فصل قبل برای سامانه پایش شبکه‌های کامپیوتری را می‌توان در چهار دسته کلی قرار داد:

- هسته SNMP
- واسط کاربری
- سمت سرور
- ذخیره‌سازی اطلاعات

در ادامه برای هر دسته توضیحاتی ارائه خواهد شد. این توضیحات شامل ماژول‌های دربرگیرنده آن، بررسی راه‌های ممکن برای پیاده‌سازی هر ماژول و درنهایت نحوه پیاده‌سازی آن ماژول خواهد بود.

۴-۱ هسته SNMP

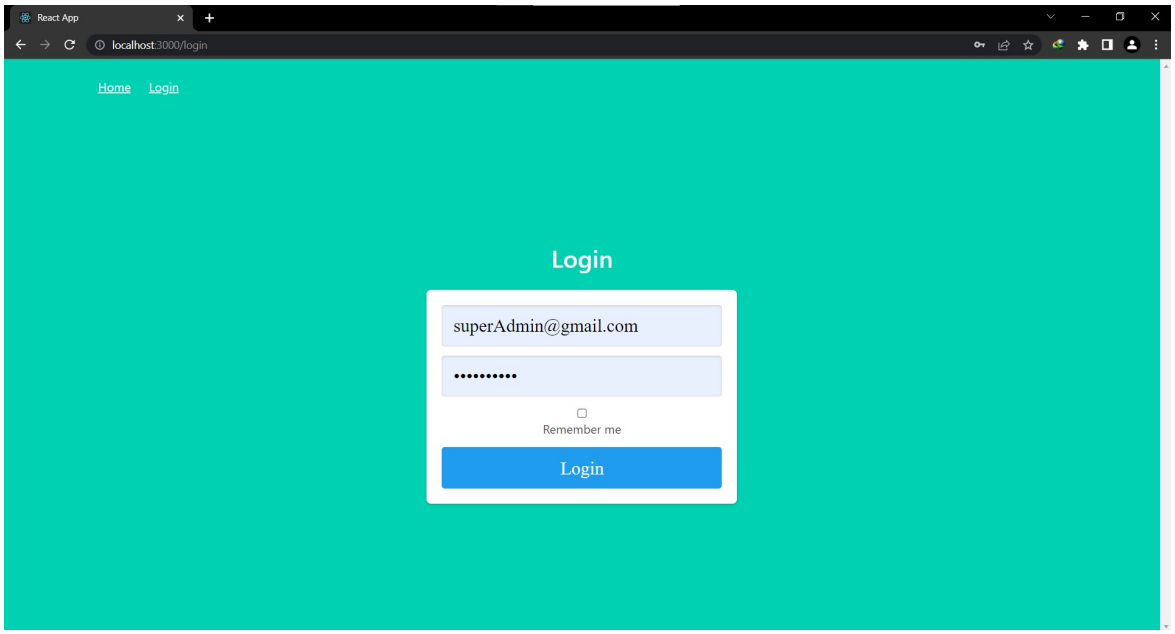
این دسته فقط شامل ماژول هسته SNMP از شکل ۳-۱ است. در این ماژول، هدف توسعه ابزاری است که بتوان از طریق آن انواع پیام‌های SNMP را ارسال و دریافت کرد. بدین منظور با تعریف دو کلاس تله[1] و نشست[2] این ماژول را پیاده‌سازی می‌کنیم. توجه شود که کتابخانه net-snmp به زبان C است ولی این ماژول جهت توسعه بهینه در زبان C++ توسعه داده شد. در نهایت ابزاری برای این ماژول تولید شد که قابلیت دریافت پیام‌های تله و همچنین ارسال و دریافت پیام‌هایی از نوع get و walk را دارد[۱۴].

[1] Trap
[2] Session

۴-۲ واسط کاربری

این دسته فقط شامل ماژول واسط کاربری تحت وب از شکل ۳-۱ است. این قسمت برای توسعه بهینه همانطور که در فصل قبل گفته شد با چارچوب ری‌اکت توسعه داده شد. در این قسمت قدم به قدم واسط کاربری توسعه داده شده به همراه کارایی آن بررسی می‌شود[۱۴].

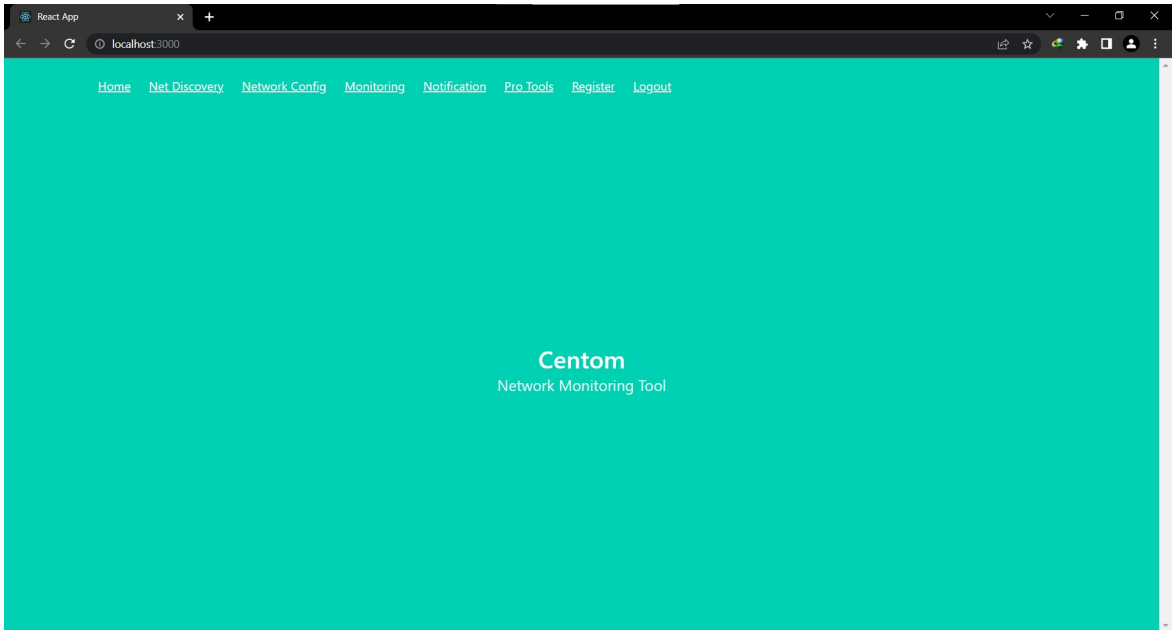
در شکل ۴-۱ و شکل ۴-۲ فهرست امکانات سامانه نشان داده می‌شود که شرح مختصری از آن‌ها در ادامه آورده شده است:



شکل ۴-۱: فهرست امکانات سیستم هنگام ورود

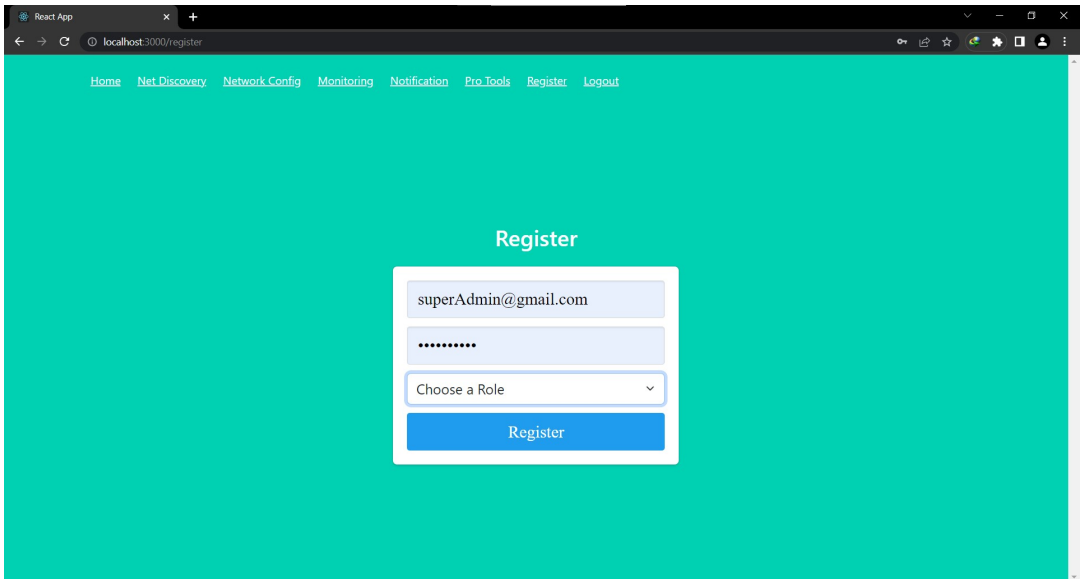
زمانی که کاربری وارد سیستم نشده است (شکل ۴-۱)، تنها دو صفحه خانه و ورود به سامانه قابل دسترسی خواهد بود. این بدان علت است که در این سامانه ثبت‌نام[1] کاربر فقط توسط مدیر ارشد صورت می‌گیرد. در شکل ۴-۲ فهرست امکانات پس از ورود مدیر ارشد نمایش داده شده‌اند. تاکید بر روی مدیر ارشد بدین علت است که در این سامانه سه سطح دسترسی مدیر ارشد، مدیر معمولی و کاربر عادی وجود دارد. مدیر ارشد به همه امکانات دسترسی دارد. همچنین مدیر معمولی به ثبت‌نام کاربر جدید، تنظیمات و اسکن شبکه دسترسی ندارد. کاربر عادی نیز فقط به اسکن سریع یک دستگاه دسترسی خواهد داشت. البته لازم به ذکر است که سامانه یک مدیر ارشد دارد!

[1] Register



شکل ۴-۲: فهرست امکانات سیستم بعد از ورود مدیر ارشد

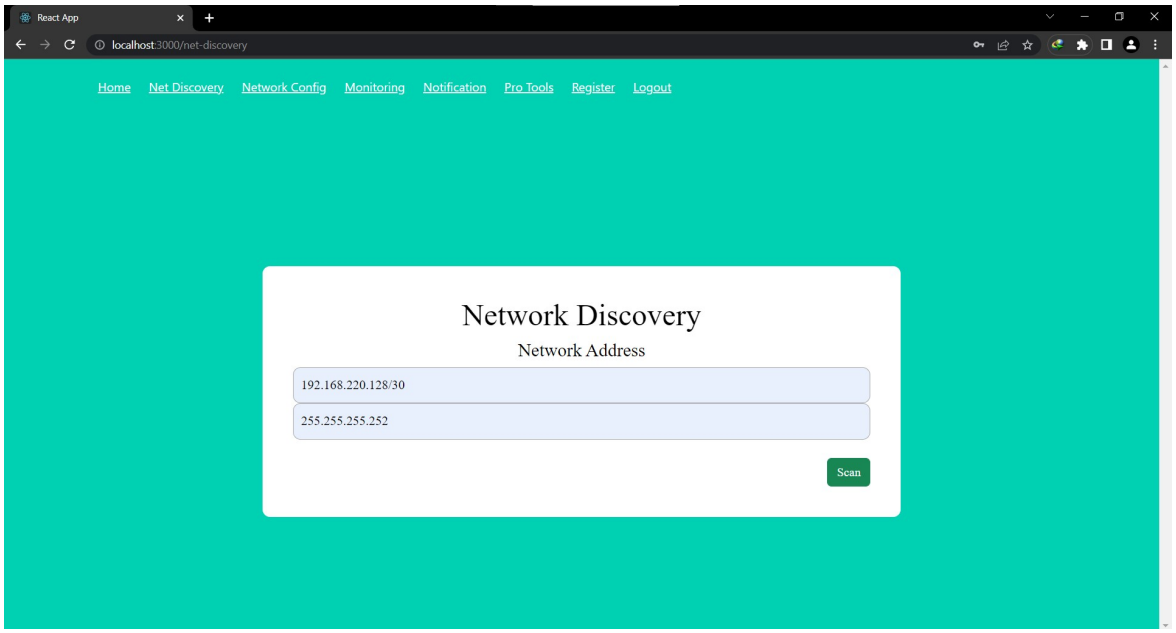
تعریف کاربر جدید فقط توسط مدیر ارشد تحت صفحه ثبت‌نام در شکل ۴-۳ ممکن خواهد بود. برای ثبت‌نام یک کاربر، مدیر ارشد ابتدا باید یک پست الکترونیکی[1] و رمز عبور وارد نماید. سپس باید برای سطح دسترسی بین دو نقش مدیر معمولی و یا کاربر معمولی انتخاب نماید. در نهایت نیز پست الکترونیکی و رمز عبور را در اختیار شخص متقاضی قرار می‌دهد.



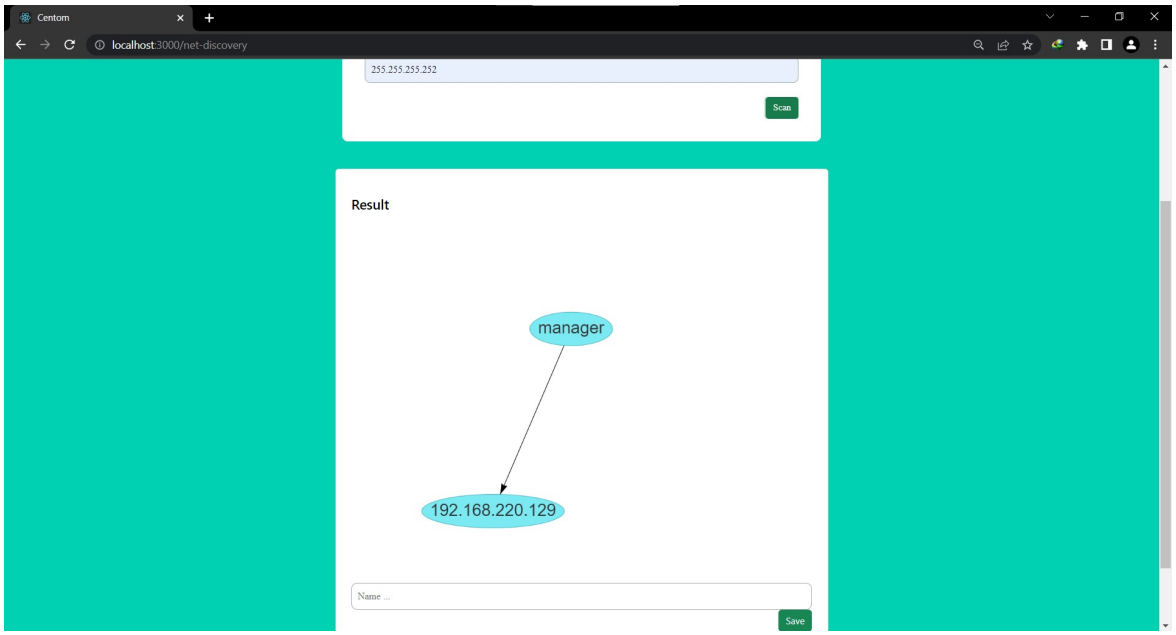
شکل ۴-۳: صفحه ثبت‌نام در اختیار مدیر ارشد

[1] Email

مهم‌ترین قسمت این سامانه، کشف شبکه است که اولین گزینه بعد از صفحه خانه قرار دارد. واسط کاربری این قسمت قبل و بعد از تست در شکل ۴-۴ و شکل ۴-۵ نشان داده شده است.



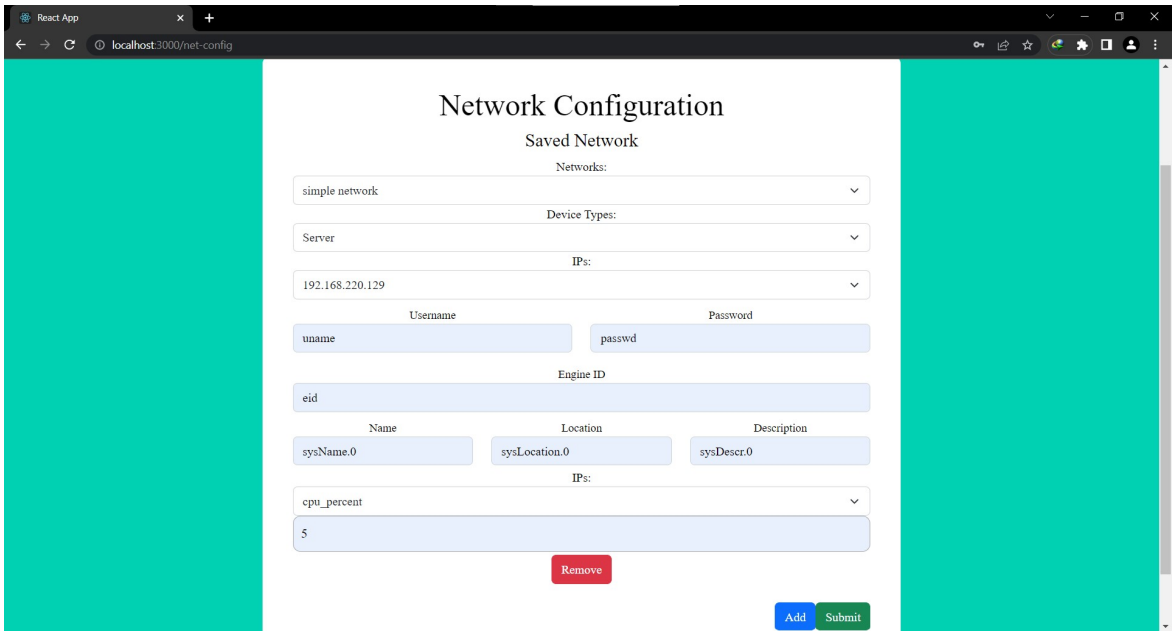
شکل ۴-۴: صفحه کشف شبکه قبل از اسکن



شکل ۴-۵: خروجی اسکن شبکه در قالب یک گراف

در کشف شبکه ابتدا یک آدرس شبکه به همراه نقاب زیر شبکه[1] از مدیر ارشد گرفته می‌شود. سپس بعد از اتمام اسکن شبکه، خروجی در قالب یک گراف نمایش داده می‌شود. درنهایت مدیر ارشد می‌تواند در صورت نیاز شبکه را با اسمی دلخواه در سامانه ذخیره کند.

بعد از ذخیره سازی شبکه در قسمت قبل، مدیر ارشد می‌تواند ابتدا اطلاعات و تنظیمات شبکه را وارد نماید و بعد از آن اقدام به پایش شبکه کند. بدین ترتیب بعد از صفحه کشف شبکه، صفحه ذخیره تنظیمات شبکه در شکل ۴-۶ وجود دارد. ابتدا با انتخاب اسم شبکه و نوع دستگاه‌های مورد نظر سامانه لیستی از آدرس‌ها را نشان می‌دهد. که با انتخاب یک آدرس مشخصات آن توسط کاربر وارد می‌شود. همچنین می‌تواند پارامترهای مورد نظر برای پایش دستگاه را به صورت لیست اضافه کند.

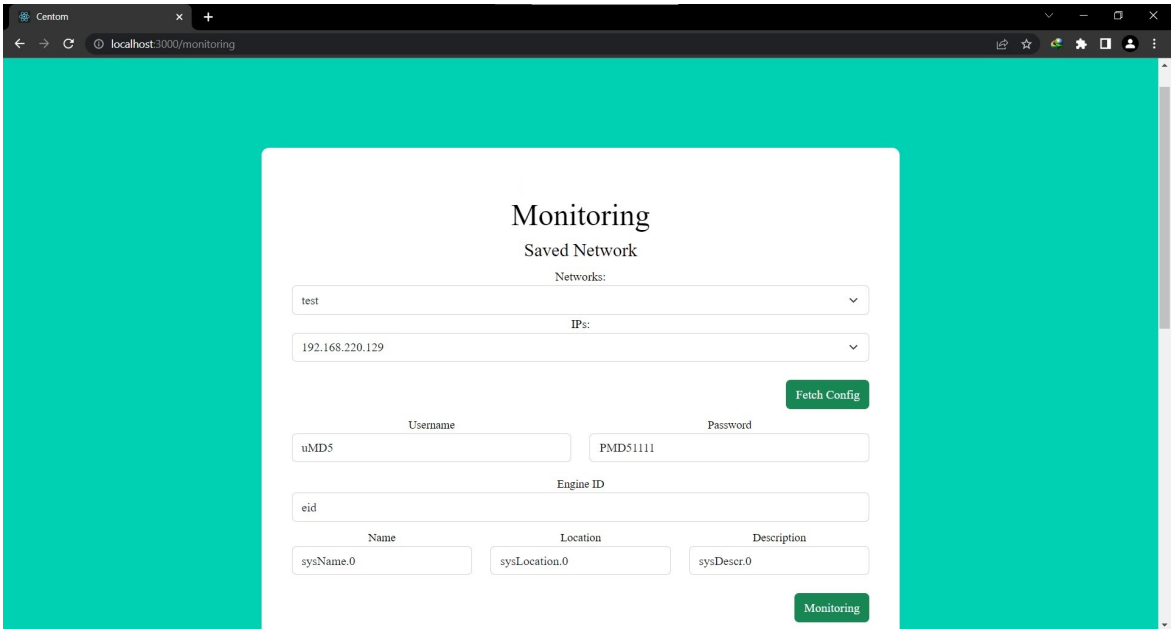


شکل ۴-۶: صفحه ذخیره تنظیمات شبکه

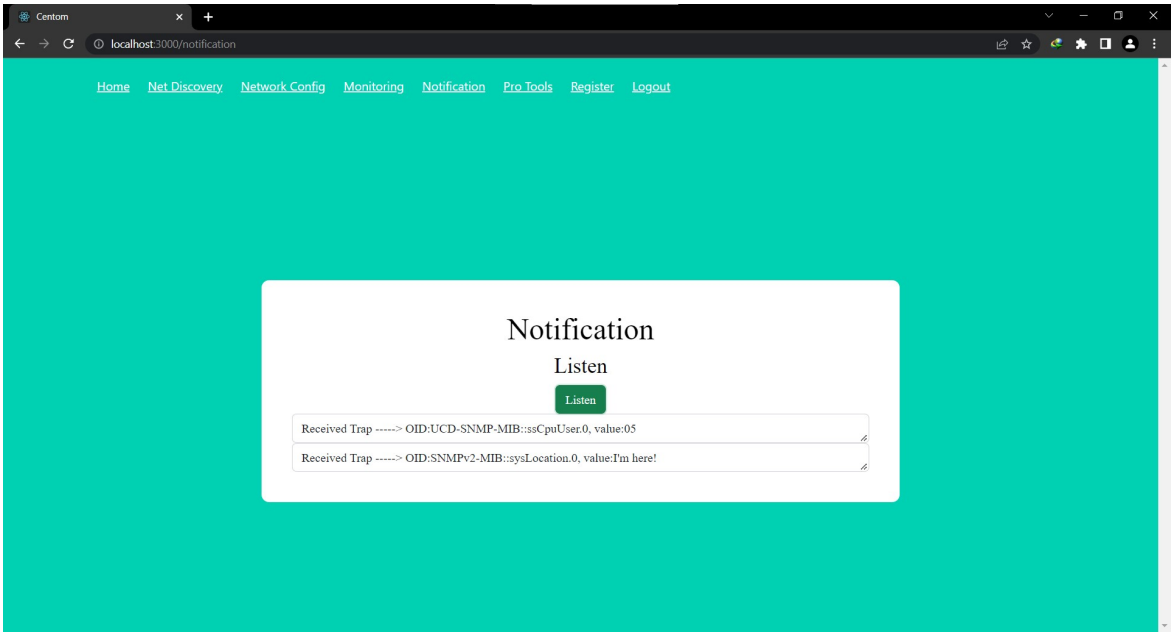
بعد از دریافت تنظیمات شبکه در مرحله قبل حال نوبت به پایش شبکه مورد نظر می‌رسد. این امر در صفحه پایش شبکه طبق شکل ۴-۷ محقق می‌شود. در این صفحه بعد از انتخاب شبکه دلخواه و یک آدرس مشخص، سامانه اقدام به پایش عملکرد دستگاه مورد نظر خواهد کرد.

بعد از گزینه صفحه پایش، طبق معماری سامانه، صفحه اعلانات طبق شکل ۴-۸ قرار دارد. در این صفحه پس از فشردن دکمه مورد نظر، سامانه به پورت ۱۶۲ گوش می‌دهد و تله‌های دریافتی را در قالبی خوانا به مدیر نمایش می‌دهد.

[1] Subnet Mask

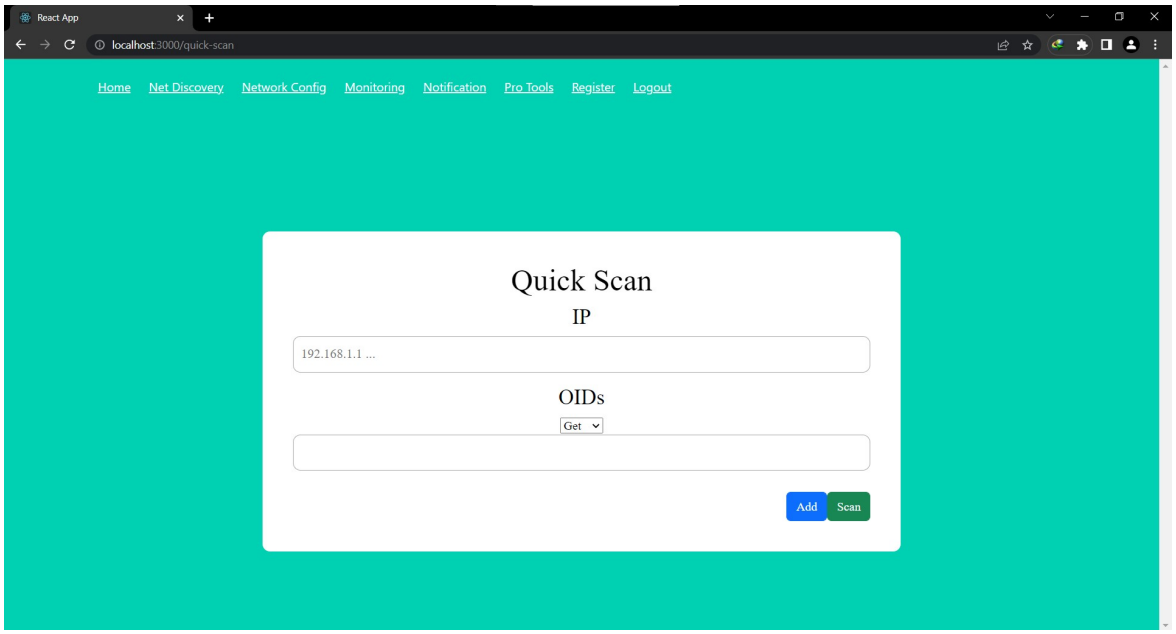


شکل ۴-۷: صفحه پایش شبکه



شکل ۴-۸: صفحه اعلانات دریافتی از شبکه

در نهایت صفحه ابزارهای پیشرفته قرار دارد، که در حال حاضر تنها اسکن سریع آن در شکل ۴-۹ موجود است. سامانه در این قسمت می‌تواند با دریافت یک آدرس و یک شناسه شی[1]، پیام‌های get و walk را ارسال و نتیجه را به ترتیب مشاهده کند.



شکل ۴-۹: صفحه ابزارهای پیشرفته

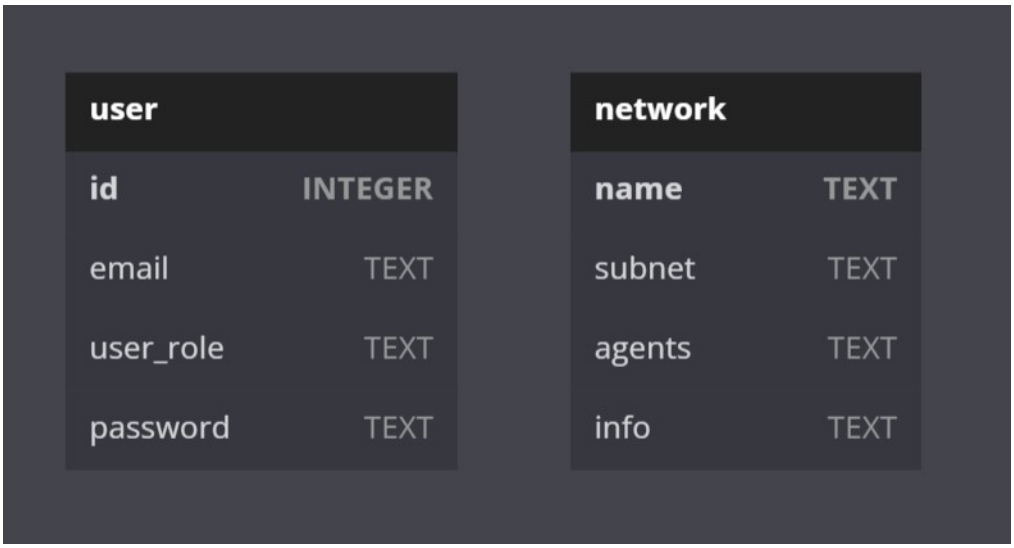
[1] Object Identifiers

۴-۳ ذخیره‌سازی اطلاعات

این دسته فقط شامل ماژول ذخیره‌سازی اطلاعات شبکه از شکل ۳-۱ است. در این سامانه به طور کلی سه نوع داده کاربران، اطلاعات شبکه و اطلاعات دریافتی از عناصر تحت مدیریت شبکه وجود دارد[۱۴].

۴-۳-۱ ذخیره‌سازی اطلاعات شبکه و کاربران

برای ذخیره‌سازی اطلاعات کاربران و اطلاعات شبکه از پایگاه داده SQLite با تعریف جدول‌ها شکل شکل ۴-۱۰ استفاده می‌کنیم.



شکل ۴-۱۰: تصویر جدول‌های تعریف شده در پایگاه داده SQLite

فیلدهایی که نیاز است توضیحی درباره آن‌ها داده شود به صورت زیر است:

- user-role: نقش هر کاربر جهت دسترسی‌های متنوع به سامانه را می‌دهد. به عنوان مثال مدیر ارشد به همه امکانات، مدیر معمولی به همه امکانات به جز ثبت‌نام کاربر جدید، تنظیمات و اسکن شبکه دسترسی دارند. و در نهایت کاربر عادی فقط به اسکن سریع یک دستگاه دسترسی خواهد داشت.
- name: این فیلد فقط برای شناسایی متمایز شبکه‌ها کاربرد دارد. بدین صورت که با دریافت یک اسم از کاربر به هنگام ذخیره، تنها زمان بازیابی اطلاعات شبکه در سامانه کاربرد دارد.

- agents: این فیلد شامل لیستی از دستگاه‌های شبکه است که عامل SNMP بر روی آن‌ها فعال است.
- info: این فیلد شامل شی‌ای از JSON[1] شامل گره‌ها، یال‌ها و نوع دستگاه‌های تحت مدیریت است.

برای اتصال به پایگاه داده SQLite نیز از SQLAlchemy که یک ابزار نگاشت رابطه به شی[2] است، استفاده شد که مزایای بسیاری دارد.

۴-۳-۲ ذخیره‌سازی اطلاعات دریافتی از شبکه

این اطلاعات در درجه اول شامل پارامترهای مختلف جهت پایش دستگاه‌ها به همراه نرخ نمونه‌برداری[3] آن‌ها است. در درجه بعد شامل اطلاعات دریافتی از دستگاه‌های تحت مدیریت مربوط به پارامترهای مختلف خواهد بود. به علت تعداد بالای بازیابی این گونه اطلاعات از ردیس استفاده شد. برای اتصال به ردیس نیز از ماژول ردیس در پایتون استفاده شد.

[1] JavaScript Object Notation
[2] Object–relational mapping (ORM)
[3] rate

۴-۴ سمت سرور

این دسته شامل ماژول‌های پردازش اطلاعات شبکه، هشدار، کشف شبکه و جمع‌آوری اطلاعات از شکل ۳-۱ است. ادامه این فصل به بررسی ماژول‌های گفته شده می‌پردازد.

۴-۴-۱ ماژول کشف شبکه

در این ماژول با توجه به نیازمندی‌های گفته شده در فصل تحلیل و طراحی، سامانه باید قادر باشد تا با دریافت یک آدرس شبکه، عناصری که عامل SNMP بر روی آن‌ها فعال هستند را به همراه نوع عنصر (سرور، مسیریاب، سوییچ و تکرارکننده) آن‌ها مشخص کند. این خروجی باید بر اساس یک توپولوژی شبکه باشد.

این مسئله با توسعه یک الگوریتم مشخص باید حل شود. در ادامه تکنیک‌های موجود برای حل این مسئله بیان و بررسی می‌شوند. در نهایت نیز راه حل به کار گرفته شده ارائه می‌شود.

تکنیک‌های حل این مسئله:

- استفاده از پینگ[1] همه‌پخشی[2] برای شناسایی تمام عناصر شبکه
- استفاده از پروتکل‌های کشف لایه پیوند[3] و یا کشف سیسکو[4]
- استفاده از پیام SNMP با شناسه شی sysServices.0

به علت عدم پشتیبانی بعضی از دستگاه‌ها از پروتکل‌های کشف لایه پیوند و کشف سیسکو و نیاز به ابزارهایی اضافی جهت حل این مسئله، از این دو پروتکل استفاده نشد. اما الگوریتمی که برای این مسئله توسعه داده شد به شرح زیر می‌باشد[۱۴]:

۱. ارسال یک پیام SNMP با شناسه شی sysServices.0 به تمام آدرس‌های موجود در آدرس شبکه (اگر پاسخی دریافت شود یعنی عنصر نیازمند مدیریت است.)

۲. رمز گشایی مقدار مرحله قبل در صورت دریافت پاسخ از آدرس مربوطه به صورت زیر[۱۵]:

[1] Ping
[2] Broadcast
[3] Link Layer Discovery Protocol (LLDP)
[4] Cisco Discovery Protocol (CDP)

(آ) تبدیل عدد برگردانده شده به فرمت باینری و تعیین نوع عنصر بر اساس مقادیر بیت‌ها (کم ارزش‌ترین بیت، بیت اول در نظر گرفته می‌شود)

(ب) اگر بیت اول تنظیم شده باشد، دستگاه موردنظر یک نوع تکرارکننده است.

(ج) اگر بیت دوم تنظیم شده باشد، دستگاه موردنظر یک نوع سوییچ است.

(د) اگر بیت سوم تنظیم شده باشد و همچنین مقدار پیام SNMP با شناسه شی ipForward-ing.0 یک باشد، دستگاه موردنظر یک نوع مسیریاب است.

(ه) اگر هیچ کدام از موارد بالا نباشد و همچنین بیت چهارم یا هفتم تنظیم شده باشد، دستگاه موردنظر یک نوع سرور است.

(و) اگر مقدار برگردانده شده در موارد بالا صدق نمی‌کرد، نوع دستگاه متفرقه[1] خواهد بود (مثل یک دستگاه منبع تغذیه اضطراری[2]).

۳. به ازای تمام آدرس‌هایی که عامل SNMP بر روی آن‌ها در حال اجرا هستند، دستور traceroute اجرا می‌شوند. خروجی بدین صورت خواهد بود که تا رسیدن به مقصد نهایی گام‌های میانی نمایش داده می‌شوند. بدین ترتیب به تعداد عناصر فعال، مسیرهای رسیدن تا آن‌ها بدست می‌آیند.

۴. در نهایت برای رسم توپولوژی شبکه تحت یک گراف، با داشتن گره‌ها و همچنین یال‌های بدست آمده از مسیرها این امر ممکن می‌شود.

۴-۴-۲ ماژول پردازش اطلاعات شبکه

در این ماژول اطلاعات دریافت شده از سمت دستگاه‌های شبکه یعنی همان پیام‌های SNMP، با توجه به مقدار آن‌ها پردازش می‌شوند. این پردازش شامل دو قسمت یعنی ابتدا نوع داده دریافتی را مشخص کرده و بعد از در صورت نیاز به حذف تعدادی کاراکتر از آن، اقدام می‌شود. قسمت دوم این پردازش بدین جهت خواهد بود که به عنوان مثال بعضی مقادیر حجم حافظه و ... در انتها عبارت KB وجود دارد.

[1] Other

[2] Uninterruptible Power Supply (UPS)

۴-۴-۳ ماژول جمع‌آوری اطلاعات

برای راحتی انجام پایش و تنظیمات شبکه توسط سامانه، در پروژه یک فایل با پسوند JSON طبق شکل ۴-۱۱ تعبیه شده است. در این فایل دو نوع کلید وجود دارد. اولین کلید، پارامترهای پیش‌فرض[1] است. اهمیت بعضی پارامترها باعث اضافه شدن این بخش شد تا این پارامترها برای تمام دستگاه‌ها بدون نیاز به افزودن توسط مدیر ارشد رصد شوند. در ادامه هر کلید متعلق به این کلید، برای یک پارامتر به خصوص شامل شناسه شی، نرخ نمونه‌برداری و نحوه پردازش آن برای نمایش است. ساختار زیرین دومین کلید یعنی پارامترها[2] نیز به همین صورت است. با این تفاوت که به مدیر ارشد لیستی از این موارد نشان داده شده و او می‌تواند برای هر آدرس در بخش تنظیمات شبکه از این لیست انتخاب نماید. در ادامه نیز برای ارسال اطلاعات به صورت بی‌درنگ[3] از SSE[4] استفاده شد. که توسعه آن در فلسک و ری‌اکت کمی با چالش نیز همراه بود[۱۴].

۴-۴-۴ ماژول هشدار

ماژول هشدار که در قالب کلاس تله وظیفه خود را انجام می‌دهد، وظیفه دریافت اطلاعات و تفسیر آن‌ها را برعهده دارد. همچنین اشاره به این نکته که تمام پیام‌های SNMP با asn.1[5] کدگذاری[6] و ارسال می‌شوند، لازم است[۱۶]. در واقع کاری که کلاس تله انجام می‌دهد را می‌توان به صورت مراحل زیر بیان کرد[۱۴]:

۱. گوش دادن به پورت ۱۶۲ و دریافت اطلاعات

۲. تبدیل اطلاعات در قالب هگزادسیمال[7] به فرمت رشته[8] در زبان C++

۳. حذف کاراکترهای فاصله در رشته

۴. کدگشایی[9] رشته مورد نظر با asn.1 از طریق ابزارهای xxd و openssl

[1] def_params
[2] params
[3] real-time
[4] Server-Sent Events
[5] Abstract Syntax Notation One
[6] encode
[7] hexadecimal
[8] String
[9] decode

شکل ۴-۱۱: تصویر فایل JSON جهت مدیریت پارامترها

۴-۵ خلاصه

در این فصل با توجه به معماری طراحی شده در فصل قبل، با گروه‌بندی ماژول‌ها، نحوه پیاده‌سازی هر ماژول بیان شد. در ابتدا نحوه پیاده‌سازی هسته SNMP بیان شد. بعد از آن به توضیح پیاده‌سازی واسط کاربری پرداخته شد. در ادامه نیز ذخیره‌سازی اطلاعات و درنهایت سمت سرور توضیح داده شدند. بخش سمت سرور نیز خود از چندین ماژول پردازش اطلاعات شبکه، هشدار، کشف شبکه و جمع‌آوری اطلاعات تقسیم شده و به هرکدام به تفکیک پرداخته شد.

فصل پنجم

تست و بررسی سامانه

بعد از فصل پیاده‌سازی، این فصل به تست و بررسی سامانه می‌پردازد. برای تست سامانه، موارد آزمون[1] برای دو دسته نیازمندی‌های کارکردی و غیرکارکردی نوشته شده و هریک به صورت جداگانه بررسی می‌شود. اما در بعضی موارد به دلیل فرصت کم به بررسی نیازمندی سامانه بسنده می‌شود. شرایط تست و بررسی نیازمندی‌های مختلف نیز به صورت زیر است:

- نوع پردازنده مرکزی[2] سیستم‌ها از نوع Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz است.
- سیستم شامل سامانه پایش شبکه و سیستم‌های دیگر جهت تست کشف شبکه، هر کدام دارای دو هسته مجازی هستند.
- سیستم شامل سامانه پایش شبکه دارای چهار گیگابایت حافظه اصلی[3] و سیستم‌های دیگر جهت تست کشف شبکه، هر کدام دارای دو گیگابایت حافظه اصلی هستند.

در ادامه در بخش‌های تست نیازمندی‌های کارکردی و غیرکارکردی به موارد آزمون یا بررسی نیازمندی‌های مختلف پرداخته می‌شود.

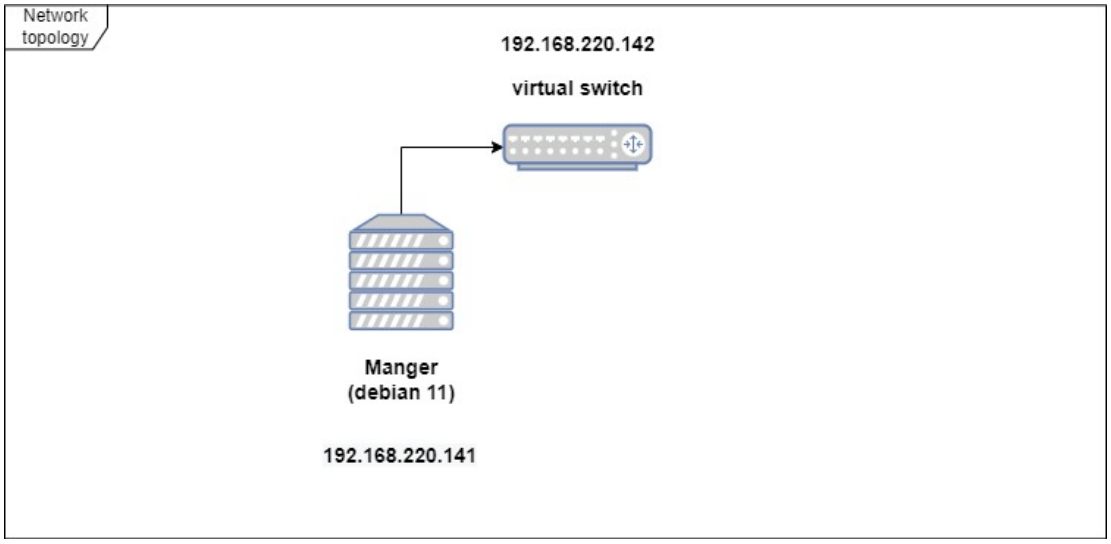
۵-۱ تست و بررسی نیازمندی‌های کارکردی

۵-۱-۱ تست کشف شبکه

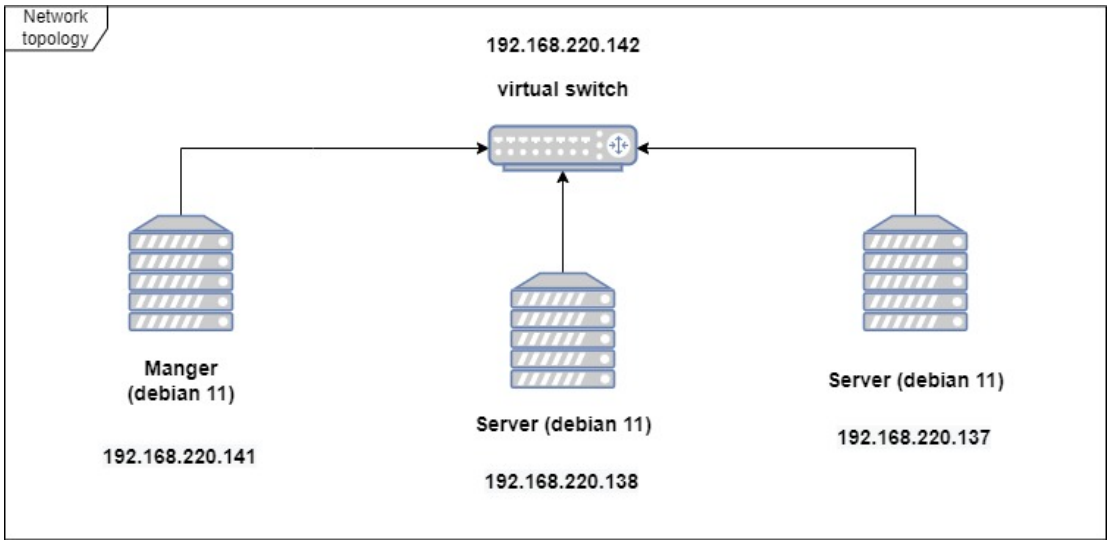
برای تست کشف شبکه با استفاده از ماشین‌های مجازی[4] و سوئیچ‌های مجازی[5]، برای دو سناریو مورد آزمون تولید شد. به عبارتی دو شبکه طبق شکل ۵-۱ و شکل ۵-۲ طراحی شد. سپس با استفاده از قابلیت کشف شبکه سامانه خروجی این دو سناریو دریافت شد، که در شکل ۵-۳ و شکل ۵-۴ قابل مشاهده است.

در شکل ۵-۱ هدف پایش تنها یک دستگاه بود. در واقع با راه‌اندازی یک سیستم و سامانه پایش برروی همان سیستم، ماژول کشف شبکه سامانه باید بتواند خودش را به درستی شناسایی کند که این امر در شکل ۵-۳ محقق می‌شود.

[1] Test case
[2] Central Processing Unit (CPU)
[3] Random-access memory (RAM)
[4] Virtual Machine (VM)
[5] Virtual Switch

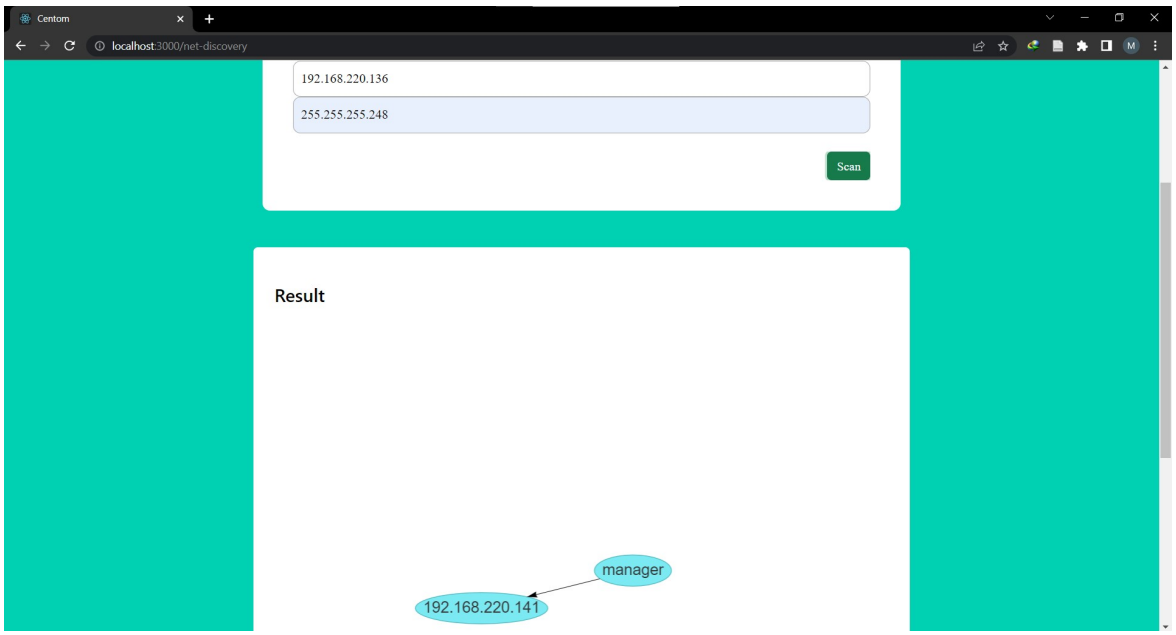


شکل ۵-۱: تصویر توپولوژی شبکه اول جهت تست

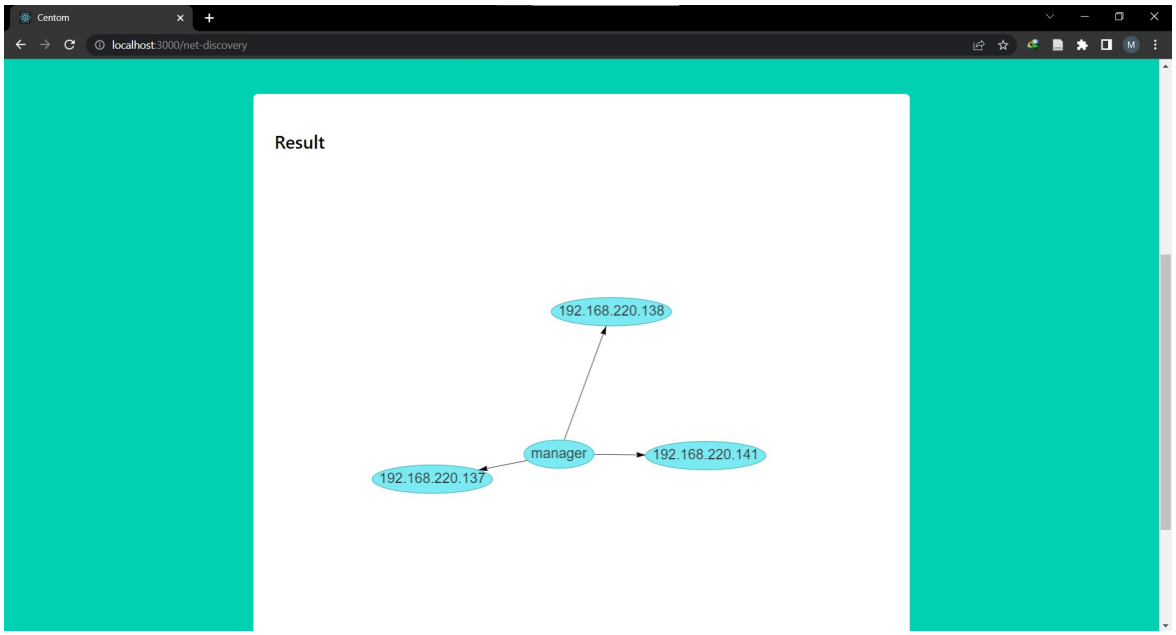


شکل ۵-۲: تصویر توپولوژی شبکه دوم جهت تست

بعد از انجام تست اول، به سوئیچ موجود در شبکه اول دو دستگاه مشابه دستگاه موجود، به شبکه اضافه می‌شوند. بعد از گرفتن خروجی شبکه دوم، مشاهده و واضح می‌شود که سوئیچ موجود در شبکه‌ها در خروجی وجود ندارد، که این امر بدین علت است که از سوئیچ مجازی برای تست استفاده شده است. خروجی شبکه دوم نیز در شکل ۵-۴ نیز قابل مشاهده است که با شبکه طراحی شده مطابقت می‌نماید.



شکل ۵-۳: تصویر توپولوژی بدست آمده برای شبکه اول



شکل ۵-۴: تصویر توپولوژی بدست آمده برای شبکه دوم

۵-۱-۲ تست تنظیمات شبکه

برای این بخش امکانی فراهم شده تا کاربر هنگام پایش اطلاعات شبکه، مواردی که در بخش تنظیمات ذخیره کرده است را مشاهده می‌کند. بدین صورت که کاربر ابتدا باید در صفحه پایش، تنظیمات دستگاه موردنظر را واکشی کند تا از صحت آن مطمئن شود. موارد بسیار زیادی در این بخش تست شد که به عنوان نمونه می‌توان به شکل ۴-۷ اشاره کرد.

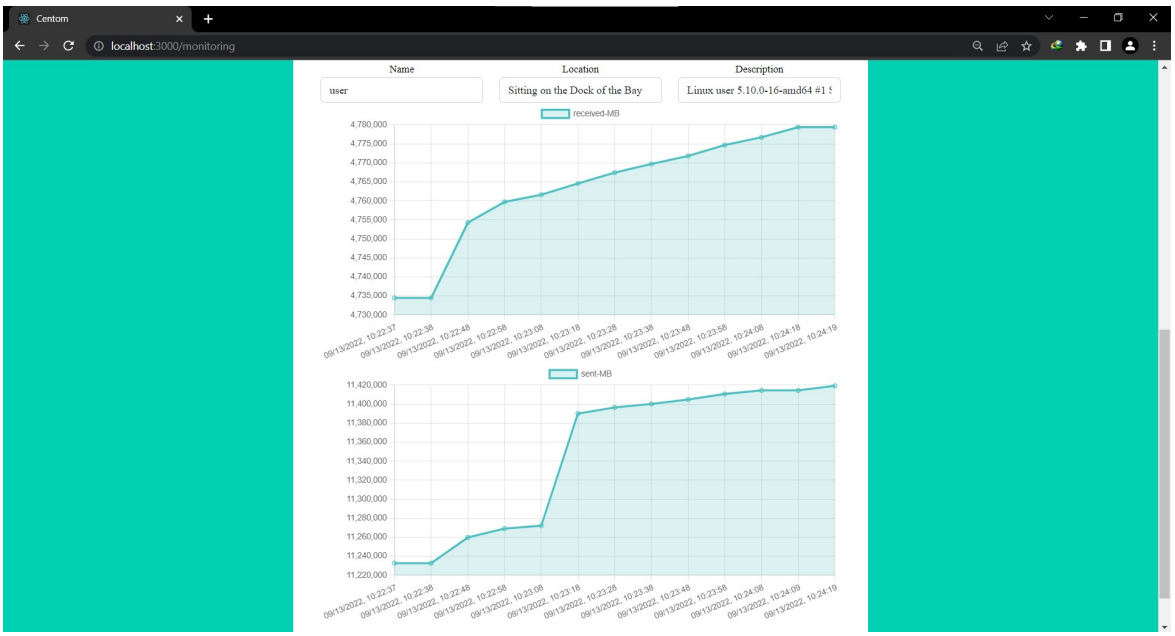
۵-۱-۳ تست پردازش و ذخیره اطلاعات جمع‌آوری شده

این تست از دو بخش تشکیل می‌شود:

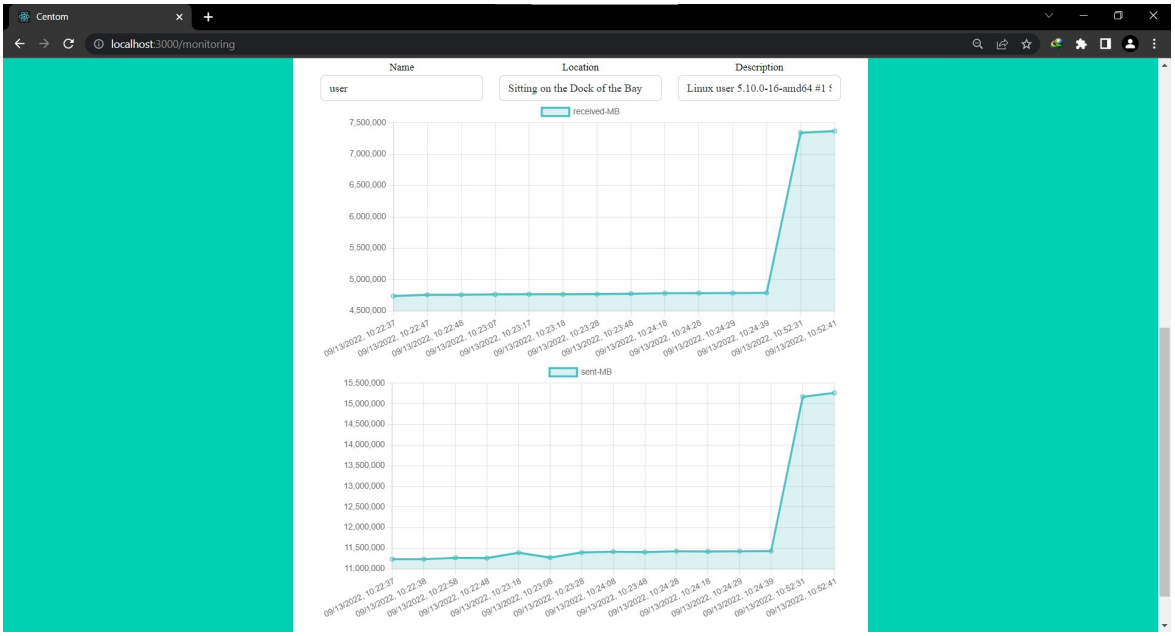
- نمایش نمودارهای پایش عملکرد عنصر تحت مدیریت
- نمایش اطلاعات قدیمی در صورت پایش قبلی

برای بخش اول شکل ۵-۵ کفایت می‌کند، چون نمودارها به صورت بی‌درنگ درحال نمایش اطلاعات دریافتی هستند.

برای بخش دوم نیز دوباره بعد از بستن صفحه موردنظر دوباره اقدام به پایش کرده که نتیجه آن در شکل ۵-۶ قابل مشاهده است. درواقع اطلاعات قدیمی از ردیس بازیابی شده و نمایش داده می‌شود. همانطور که در شکل ۵-۶ دیده شد، پرش مربوط در نمودارهای ارسال و دریافت برای مدت زمانی است که پایشی صورت نگرفته است اما از طریق سیستم اطلاعات ارسال و دریافت شده است.



شکل ۵-۵: تصویر نمودارهای مربوط به پارامترها



شکل ۵-۶: تصویر نمودارهای پارامترها جهت نمایش بازیابی صحیح اطلاعات

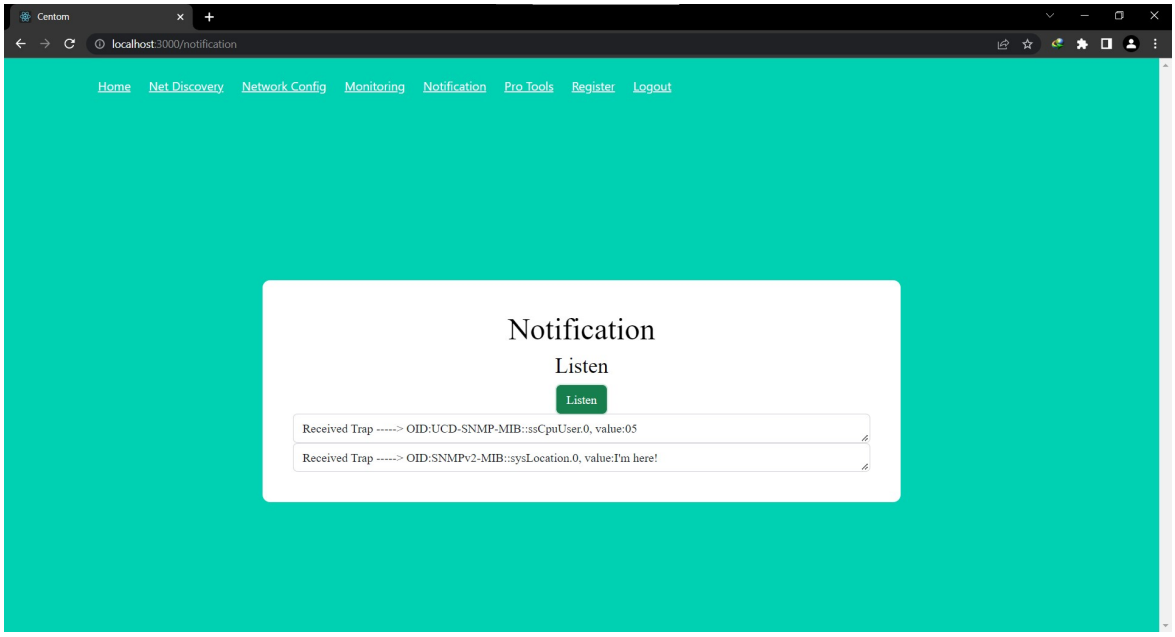
۵-۱-۴ تست جمع‌آوری هشدارهای مربوط به عناصر تحت مدیریت

برای این بخش از ابزار snmptrap که تله‌ای مشخص می‌فرستد، استفاده شد. با این ابزار تله‌هایی طبق شکل ۵-۷ به سامانه ارسال کرده و باید نتیجه در صفحه اعلانات بعد از فشردن دکمه مشاهده شود (به همان ترتیب و مقادیر)

همان‌طور که طبق شکل ۵-۸ مشاهده می‌شود نتایج دقیقا با دستورات اجرایی مطابقت می‌نماید.

```
root@user:/home/user/Centom# snmptrap -v 2c -c public localhost "" linkUp.0 ssCpuUser.0 i 5
root@user:/home/user/Centom# snmptrap -v 2c -c public localhost "" linkUp.0 sysLocation.0 s "I'm here!"
root@user:/home/user/Centom#
```

شکل ۵-۷: تصویر دستورات اجرا شده تله



شکل ۵-۸: تصویر تله‌های دریافتی در سامانه

۵-۲ تست و بررسی نیازمندی‌های غیرکارکردی

۵-۲-۱ تست مدت زمان کشف شبکه

دقیقا همان تستی که در بخش نیازمندی‌های کارکردی برای تست کشف شبکه انجام شد، کیفیت (مدت زمان) آن در اینجا درنظر گرفته می‌شود. توجه شود که در هر دو سناریو آدرس شبکه ۱۹۲.۱۶۸.۲۲۰.۱۳۶ و همچنین نقاب زیرشبکه ۲۵۵.۲۵۵.۲۵۵.۲۴۸ در نظر گرفته شدند، که در سناریوی اول زمان کشف شبکه ۳۵ ثانیه و در شبکه دوم ۳۳ ثانیه بدست آمد. این اختلاف زمانی بدین خاطر است که در سناریوی اول زمان بیشتری جهت انتظار دریافت پیغام‌های SNMP صرف شده است. البته این مقادیر کمتر از دو دقیقه (مقدار مطلوب) است.

۵-۲-۲ بررسی واسط کاربری مطلوب تحت وب

پیاده‌سازی واسط کاربری این سامانه با چارچوب ری‌اکت و همچنین چارچوب فلسک در سمت سرور، رابط کاربری تحت وب مطلوبی را برای افراد فراهم می‌آورد. همچنین برای استفاده از سامانه بیرون از شبکه سازمان، با چند تغییر در تنظیمات شبکه می‌توان به آن دسترسی پیدا کرد. این رابط کاربری طبق بررسی‌های انجام شده و مشورت‌های صورت گرفته توسعه داده شده است. همچنین پرسشنامه‌ای جهت بررسی این نیازمندی در شکل ۵-۹ طراحی شد ولی به دلیل فرصت کم امکان نظرسنجی از مدیران شبکه فراهم نشد.

نظرسنجی رابط کاربری سامانه پایش شبکه‌های کامپیوتری

نام و نام خانوادگی(اختیاری): **تاریخ:**

به موارد زیر در مقیاس ۱ تا ۵ امتیاز دهید که ۱ کاملا مخالف و ۵ کاملا موافق است.

- سامانه به راحتی قابل درک، واضح، ساده و آسان برای یادگیری است.

۱	۲	۳	۴	۵

- تعامل با محصول قابل پیش‌بینی، ایمن و مطابق انتظارات من است.

۱	۲	۳	۴	۵

- محصول نوآورانه، مبتکرانه و خلاقانه طراحی شده است.

۱	۲	۳	۴	۵

- تمایل دارم که از این سامانه از لحاظ بصری در روزمره خود استفاده کنم.

۱	۲	۳	۴	۵

- تجربه کلی خود را در استفاده از این سامانه چگونه ارزیابی می‌کنید؟

۱	۲	۳	۴	۵

- سخت ترین قسمت استفاده از این سامانه:

- هر پیشنهادی که برای بهترشدن رابط کاربری این سامانه دارید را لطفا در ادامه بنویسید:

شکل ۵-۹: پرسشنامه طراحی شده برای بررسی واسط کاربری

۵-۲-۳ بررسی امنیت سامانه

این نیازمندی به دلیل فرصت کم تنها بررسی شد و تستی برای آن صورت نگرفت. امنیت سامانه باید از سه منظر محرمانگی، صحت و در دسترس بودن بررسی شود. در ادامه برای این بخش‌ها تدابیر صورت گرفته برای توسعه این سامانه ذکر می‌شوند.

- برای فراهم آوردن محرمانگی داده‌ها از پروتکل SNMPv3 برای پایش اجزا استفاده شد که نیاز به نام کاربری و رمز عبور دارد.
- برای فراهم آوردن صحت داده‌ها استفاده از رمز عبور برای پایگاه‌های داده می‌تواند مفید باشد. به عنوان مثال برای ردیس رمز عبوری قوی بکار گرفته شد.
- در بحث در دسترس بودن سامانه کار ویژه‌ای انجام نشد، زیرا برای فراهم آوردن این ویژگی نیاز به یک فایروال در شبکه است تا به عنوان مثال از حملات منع دسترسی جلوگیری نماید.

فصل ششم

جمع‌بندی

در آخرین فصل یک جمع‌بندی از مطالب ارائه شده در این پایان‌نامه ارائه می‌شود. ابتدا خلاصه‌ای از مطالب بیان شده سپس کاربرد پروژه انجام شده و درنهایت کارهایی که درادامه این پروژه می‌توانند انجام شوند، ارائه می‌شود.

۶-۱ خلاصه

در این پروژه هدف، پیاده‌سازی یک سامانه برای پایش شبکه‌های کامپیوتری بوده است. برای تحقق این هدف، ابتدا با مروری بر روی سامانه‌های موجود، زیرمجموعه‌هایی از ویژگی‌ها استخراج شدند. پس از آن با تحلیل نیازمندی‌ها، جدول‌های نیازمندی‌های کارکردی و غیرکارکردی استخراج شدند. پس از استخراج نیازمندی‌ها جهت طراحی معماری، فناوری‌های پیاده‌سازی بررسی و برای هر بخش یک فناوری انتخاب شد. باتوجه به معماری خود فناوری‌های انتخاب شده، معماری کل سامانه طراحی شد. سپس پیاده‌سازی بر اساس ماژول‌های مختلف جهت توسعه نرم‌افزار انجام شد. درنهایت نیز تست سامانه بر اساس نیازمندی‌های تحلیل شده، انجام شد.

۶-۲ کاربردها

این پروژه کاربردهای بسیار در شبکه‌های مختلف دارد. مدیر شبکه با به کارگیری این سامانه می‌تواند عملکرد یک سیستم یا کل شبکه را پایش کند و برای اقدامات کوتاه‌مدت و بلندمدت برنامه‌ریزی کند. اقدامات کوتاه‌مدت شامل تغییر تنظیمات، تغییر مسیرهای مسیریاب‌ها و ... هستند. همچنین اقدامات بلندمدت می‌توانند شامل تهیه منابع مختلف، بهبود سرویس‌های موجود و ... باشند. همچنین هشدارهای تولید شده در سطح شبکه به راحتی قابل مشاهده خواهند بود. کاربرد دیگر این سامانه در بخش کشف شبکه خواهد بود. مدیر شبکه می‌تواند با اسکن کل شبکه، توپولوژی را به همراه نوع ماشین‌های مختلف مشاهده کند تا در نهایت شبکه را به سمت شبکه‌ای با کارایی بالا هدایت نماید.

۶-۳ کارهای آینده

در ماژول پایش شبکه مفهومی تحت عنوان نرخ نمونه‌برداری وجود دارد. این مفهوم بدین معناست که مقادیر عملکردی عناصر تحت مدیریت، با چه دوره تناوبی نمونه‌برداری شوند. در حال حاضر این مقدار توسط مدیر شبکه باید تنظیم شود. اما می‌توان این مقادیر را با استفاده از الگوریتم‌های هوش مصنوعی تحت عنوان پایش هوشمند متناسب با شبکه تنظیم کرد. این کار سه مزیت اصلی خواهد داشت:

- عدم درگیری مدیر شبکه با مقادیر نرخ نمونه‌برداری
- بهینه‌سازی مقدار ترافیک مدیریتی شبکه
- دقت بالاتر برای رسم نمودارها

از کارهای دیگر می‌توان به کاهش حجم اطلاعات در سامانه اشاره کرد. در بلندمدت حجم این اطلاعات بسیار زیاد خواهد شد. دو اقدام اصلی می‌توان جهت حل این مشکل انجام داد:

- تغییر ساختار اطلاعات به نحوی بهینه
- پردازش دوره‌ای اطلاعات و نگهداری خلاصه‌ای از آن‌ها از جمله مقادیر آماری (میانگین، انحراف‌معیار و ...)

منابع و مراجع

[1] Mauro, Douglas and Schmidt, Kevin. Essential SNMP: Help for System and Network Administrators. ” O’Reilly Media, Inc.”, 2005.

[2] Hare, Chris. Simple network management protocol (snmp)., 2011.

[3] SNMP introduction tutorial (simple network management protocol). https://www.thegeekstuff.com/2012/09/snmp-introduction/.

[4] Solarwinds documentation. https://documentation.solarwinds.com/en/success_center/ipmonitor/content/introduction.htm.

[5] What is datadog? definition from SearchITOperations. https://www.techtarget.com/searchitoperations/definition/Datadog.

[6] Olups, Rihards. Zabbix 1.8 network monitoring. Packt Publishing Ltd, 2010.

[7] Bass, Len, Clements, Paul, and Kazman, Rick. Software architecture in practice. Addison-Wesley Professional, 2003.

[8] stackoverflow. Stack overflow developer survey 2021. https://insights.stackoverflow.com/survey/2021/?utm_source=social-share&utm_medium=social&utm_campaign=dev-survey-2021.

[9] Technologies, Mindmajix. What is ReactJs - introduction to ReactJS and its features. https://mindmajix.com/introduction-to-react-js. Section: Looker.

[10] An overview of angular. in this article, we are going to have a… | by m□ty□s lancelot bors | medium. https://medium.com/@mlbors/an-overview-of-angular-3ccd2950648e.

[11] Django introduction - learn web development | MDN. https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction.

[12] Introduction to node.js. https://nodejs.dev/en/learn/.

[13] Vychodil, Hynek-Pichi. Answer to "which language and lib is better suited for high-performant network devices polling server (SNMP)?". https://stackoverflow.com/a/30896305.

[14] Mirfendereski, Mahdi. Centom. https://github.com/smmir-cent/Centom, September 2022.

[15] How are device type values selected for devices in performance management. https://knowledge.broadcom.com/external/article/32164/how-are-device-type-values-selected-for.html.

[16] Dorlan, Peter L. An introduction to computer networks. Autoedición, 2016.

واژه‌نامه‌ی فارسی به انگلیسی

آ

اتصال داده دو طرفه . . . . . Two-way Data Binding

ارلنگ . . . . . Erlang

اعلان . . . . . Notification

انگولار . . . . . Angular

ب

بی‌درنگ . . . . . Real-time

پ

پایتون . . . . . Python

پایش . . . . . Monitoring

پایگاه‌داده در حافظه اصلی . . . . In-Memory Database

پایگاه‌داده رابطه‌ای . . Relational Database

پردازنده مرکزی . Central Processing Unit (CPU)

پروتکل کشف سیسکو . . . Cisco Discovery Protocol (CDP)

پروتکل کشف لایه پیوند . . . . . Link Layer Discovery Protocol (LLDP)

پست الکترونیکی . . . . . Email

پینگ . . . . . Ping

ت

تایپ‌اسکریپت . . . . . Typescript

تله . . . . . Trap

ث

ثبت‌نام . . . . . Register

ج

جاوا اسکریپت . . . . . Javascript

جنگو . . . . . Django

چ

چارچوب . . . . . Framework

ح

حافظه اصلی . . . Random-access memory (RAM)

د

دبین . . . . . Debian

دستگاه منبع تغذیه اضطراری Uninterruptible Power Supply (UPS)

دیتاداگ . . . . . Datadog

ر

ردیس . . . . . Redis

رشته . . . . . String

روش . . . . . Methodology

ری‌اکت . . . . . React

ز

زبیکس . . . . . Zabbix

س

سرور . . . . . Server

سمت سرور . . . . . Back-end

سولارویندز . . . . . SolarWinds

سوئیچ . . . . . Switch

سوئیچ مجازی . . . . . Virtual Switch

ش

شناسه شی . . . . . Object Identifier

ف

فرآیند . . . . . Process

فلسک . . . . . Flask

فناوری . . . . . Technology

ک

کانتینر . . . . . Container

کدگذاری . . . . . Encode

کدگشایی . . . . . Decode

کشف شبکه . . . . . Network Discovery

ل

لایه کاربرد . . . . . Application layer

لینوکس . . . . . Linux

م

ماشین مجازی . . . Virtual Machine (VM)

متفرقه . . . . . Other

متن‌باز . . . . . Open-Source

مدل شیء سند مجازی . Virtual Document Object Model (Virtual DOM)

مسیریاب . . . . . Router

موتور جاوا اسکریپت وی هشت . . . . . V8 JavaScript engine

مورد آزمون . . . . . Test case

میزبان . . . . . Host

ن

نخ . . . . . Thread

نرخ نمونه‌برداری . . . . . Rate

نرم‌افزار مبتنی بر ابر . Software as a service

نشست . . . . . Session

نظرسنجی توسعه‌دهندگان استک اورفلو Stack Overflow Developer Survey

نقاب زیر شبکه . . . . . Subnet Mask

نگاشت رابطه به شی . . . Object–relational mapping (ORM)

نمایش . . . . . View

نودجی‌اس . . . . . Node.js

نیازمندی‌های غیرکارکردی Non functional requirements

نیازمندی‌های کارکردی . . . . . Functional requirements

و

ویو جی‌اس . . . . . Vue.js

ه

هگزادسیمال . . . . . Hexadecimal

همه‌پخشی . . . . . Broadcast

همه‌جانبه . . . . . Full Stack

واژه‌نامه‌ی انگلیسی به فارسی

A

Angular . . . . . انگولار

Application layer . . . . . لایه کاربرد

B

Back-end . . . . . سمت سرور

Broadcast . . . . . همه‌پخشی

C

Central Processing Unit . پردازنده مرکزی (CPU)

Cisco Discovery . . . پروتکل کشف سیسکو Protocol (CDP)

Container . . . . . کانتینر

D

Datadog . . . . . دیتاداگ

Debian . . . . . دبین

Decode . . . . . کدگشایی

Django . . . . . جنگو

E

Email . . . . . پست الکترونیکی

Encode . . . . . کدگذاری

Erlang . . . . . ارلنگ

F

Flask . . . . . فلسک

Framework . . . . . چارچوب

Full Stack . . . . . همه‌جانبه

Functional . . . . . نیازمندی‌های کارکردی requirements

H

Hexadecimal . . . . . هگزادسیمال

Host . . . . . میزبان

I

متن‌باز . . . . . Open-Source

متفرقه . . . . . Other

P

پینگ . . . . . Ping

فرآیند . . . . . Process

پایتون . . . . . Python

R

حافظه اصلی . . . Random-access memory (RAM)

نرخ نمونه‌برداری . . . . . Rate

ری‌اکت . . . . . React

بی‌درنگ . . . . . Real-time

ردیس . . . . . Redis

ثبت‌نام . . . . . Register

پایگاه‌داده رابطه‌ای . . Relational Database

مسیریاب . . . . . Router

S

سرور . . . . . Server

نشست . . . . . Session

نرم‌افزار مبتنی بر ابر . Software as a service

سولارویندز . . . . . SolarWinds

پایگاه‌داده در حافظه اصلی . . . . In-Memory Database

J

جاوا اسکریپت . . . . . Javascript

L

پروتکل کشف لایه پیوند . . . . . Link Layer Discovery Protocol (LLDP)

لینوکس . . . . . Linux

M

روش . . . . . Methodology

پایش . . . . . Monitoring

N

کشف شبکه . . . . . Network Discovery

نودجی‌اس . . . . . Node.js

نیازمندی‌های غیرکارکردی Non functional requirements

اعلان . . . . . Notification

O

شناسه شی . . . . . Object Identifier

نگاشت رابطه به شی . . . Object–relational mapping (ORM)

دستگاه منبع تغذیه اضطراری Uninterruptible Power Supply (UPS)

V

موتور جاوا اسکریپت وی هشت . . . . . V8 JavaScript engine

نمایش . . . . . View

مدل شیء سند مجازی . Virtual Document Object Model (Virtual DOM)

ماشین مجازی . . . Virtual Machine (VM)

سوئیچ مجازی . . . . . Virtual Switch

ویو جی‌اس . . . . . Vue.js

Z

زبیکس . . . . . Zabbix

نظرسنجی توسعه‌دهندگان استک اورفلو Stack Overflow Developer Survey

رشته . . . . . String

نقاب زیر شبکه . . . . . Subnet Mask

سوئیچ . . . . . Switch

T

فناوری . . . . . Technology

مورد آزمون . . . . . Test case

نخ . . . . . Thread

تله . . . . . Trap

اتصال داده دو طرفه . . . . . Two-way Data Binding

تایپ‌اسکریپت . . . . . Typescript

U

Abstract

The increasing development of computer networks has made its management very important. In general, the management of a system will include monitoring components, analyzing information, and taking action to approach the goal of that system. In other words, management and especially computer network management is a permanent process including monitoring, processing, planning, and action. The purpose of this project was to develop a tool for monitoring computer networks. With the development of this system, management information including traffic information, configuration information, warnings, etc. is collected and displayed to the user. So with this feature, the user can plan and act through this information. This system receives monitoring information from components with SNMP protocol through a management interface and displays it in an understandable way for the network manager. After receiving the management information in the system, it will be processed and finally stored in the databases. Also, the web user interface fetches this management information and displays it to the user. Another possibility that was added to this system to facilitate planning and action for management was network discovery. With this feature, the user can also get an overview of the network. At the end of the project, various functional and non-functional requirements of the system were examined and tested.

Key Words:

network management, network monitoring, SNMP protocol, network discovery

Amirkabir University of Technology
(Tehran Polytechnic)

Department of Computer Engineering

B.Sc. Thesis

Implementation of Network Monitoring System

By

Seyyed Mahdi Mirfendereski

Supervisor

Dr. Masoud Sabaei

September & 2022