

به نام خدا

پروژه توسعه ای پایانی درس سیستم های توزیع شده

مهدی آقاجانلو

۴۰۰۱۳۱۰۰۲

استاد درس: جناب آقای دکتر کلباسی

زمستان ۱۴۰۱

قسمت صفر.....	۴
چارچوب تعریف مسئله.....	۴
تعریف مسئله:.....	۴
اهداف:.....	۴
پیش نیاز ها:.....	۵
چالش ها:.....	۸
قسمت اول- اتاق جلسه.....	۹
تطبیق اهداف، نیازمندی ها و چالش ها.....	۹
تطبیق اهداف.....	۹
تطبیق پیش نیاز ها.....	۹
طراحی اول.....	۱۰
پیش فرض های اختصاصی.....	۱۰
کشف سرویس و توزیع شناسه.....	۱۰
هماهنگی های صوت.....	۱۲
هماهنگی های دیگر (کنترل محتوا).....	۱۳
اجرا کننده ی محتوا.....	۱۴
توزیع کردن محتوا.....	۱۵
ارزیابی طراحی اول.....	۱۵
طراحی دوم.....	۱۷
پیش فرض های اختصاصی.....	۱۷
کشف سرویس و توزیع شناسه.....	۱۷
هماهنگی ها.....	۱۸
توزیع محتوا.....	۱۸
ارزیابی طراحی دوم.....	۱۸
مقایسه ی طراحی ها.....	۲۰
قسمت دوم- تعمیم به استادیوم.....	۲۱
قسمت سوم- محتوای تصویری.....	۲۴
صفحه مجازی.....	۲۴
چینش.....	۲۴
چینش دلخواه.....	۲۵
چینش پیشنهادی.....	۲۶
مراجع:.....	۲۷

فهرست اشکال:

- شکل ۱) ابعاد و ساختار رایج برای اتاق جلسه [2] ۹
- شکل ۲) فرمت MP3 frame به صورت شماتیک [4] ۱۴

قسمت صفر

چارچوب تعریف مسئله

در ابتدا یک چارچوب تعریف مسئله مشخص می‌شود؛ سپس این چارچوب در قسمت های مختلف پروژه با توجه به تعریف مسئله ی آن قسمت از پروژه، به روز رسانی شده و تطبیق داده می‌شود. به عبارتی دیگر این چارچوب در قسمت های مختلف متناسب با شرایط بازنویسی شده و تطبیق داده می‌شود که شامل موارد زیر خواهد بود:

- قسمت اول از پروژه برای ۱۰ کاربر در یک اتاق
- قسمت دوم از پروژه برای ۱۰۰ هزار نفر در استادیوم آزادی
- قسمت سوم از پروژه برای پخش محتوای تصویری

تعریف مسئله:

تعریف مسئله ی مطرح شده در قسمت اول پروژه ی توسعه ای، در قالب تعدادی هدف، پیش‌نیاز و چالش انجام شده است که در ادامه به بررسی جزئی هر مورد پرداخته خواهد شد.

اهداف:

اهداف در نظر گرفته شده برای سامانه را می‌توان در موارد زیر خلاصه کرد:

۱. چنین سامانه ای برای تعداد کاربران مطلوب به راحتی قابل استفاده باشد و افزایش تعداد کاربران در عملکرد سامانه تاثیر چشم گیری نداشته باشد.
۲. صدای پخش شده در این سامانه، فارغ از محل قرار گیری شنونده در محیط، بلند تر از حالت عادی باشد. (فرض می‌شود هر کاربر حداقل یک گوشی همراه خود دارد).
۳. صدای پخش شده در این سامانه، فارغ از محل قرار گیری شنونده در محیط، دارای کیفیت شنیداری کافی باشد. (کیفیت شنیداری در این‌جا، به صورت تجربه نهایی شنیداری یک انسان از یک صوت تعریف می‌شود).
۴. سادگی استفاده و شفافیت توزیع شدگی سامانه: کاربر، چه به عنوان پخش کننده ی موسیقی و چه به عنوان شنونده نباید از پیچیدگی های پیاده سازی و فرایند های پشت صحنه ی توزیع شدگی اطلاعات داشته باشد. از دیدگاه کاربر، یک برنامه وجود دارد که می‌تواند فایل صوتی را همزمان از دستگاه های مختلف پخش کند و کیفیت شنیداری را افزایش دهد. جزئیات دیگر در رابطه با این سامانه بایستی تا حد امکان از دید کاربر مخفی شوند و بسیاری از تصمیمات مورد نیاز جهت نیل به اهداف، بایستی به صورت خودکار و با پیش‌بینی نیاز کاربر گرفته شوند.

پیش نیاز ها:

سامانه جهت دستیابی به اهداف، بایستی مجموعه ای از پیش نیاز ها را به طور قطع داشته باشد. نیل به اهداف بدون نیل به پیش نیاز ها غیر ممکن خواهد بود. با این تعریف، پیش نیاز های این سامانه را می توان موارد زیر تعریف کرد:

۱. کنترل اجرا: بایستی مکانیزمی جهت انجام اعمال کنترل اجرا مانند شروع و باز سرگیری اجرا، توقف و قطع اجرا، تنظیم بلندی صدا و یا اعمال پیچیده تری همچون قابلیت seek در طول رسانه، تنظیم سرعت پخش و ... وجود داشته باشد.

۲. پشتیبانی از فرمت های صوتی مختلف: از آنجایی که فرمت های مختلف صوتی می توانند در کارایی کلی سامانه تاثیر مستقیم داشته باشند، بایستی تدبیری اندیشید که بتوان از طیف گسترده ای از فرمت های صوتی و یا تصویری در این سامانه بهره برد.

۳. سازگاری سامانه با سیستم عامل های مختلف و دستگاه های مختلف

۴. رابط کاربری مناسب: بایستی یک رابط کاربری مناسب جهت انجام اعمالی چون انتخاب پخش کننده (کسی که کنترل انتخاب رسانه، اجرا و توقف و ... رسانه را دارد)، یک رابط کاربری مشابه پخش کننده های موسیقی برای انجام اعمال مرتبط با پخش مانند توقف و اجرا و ... وجود داشته باشد.

۵. مکانیزم تخصیص نقش: در این سامانه حداقل سه نقش برای کاربر می توان متصور بود:

a. میزبان پخش: نقشی که کنترل رسانه را در اختیار دارد را گوئیم. این نقش اختیار اعمالی چون پخش، توقف، تنظیم بلندی، انتخاب رسانه و نقش های مشابه را بر عهده دارد.

b. پخش کننده: نقشی که تنها وظیفه بازپخش را بر عهده دارد را گوئیم.

c. میزبان نشست: این نقش به کسی که اقدام به ساخت نشست پخش همگانی صوت می نماید تعلق می گیرد.

در چنین وضعیتی نه تنها نباید نیاز به نصب دو برنامه مجزا برای قبول این نقش ها باشد، بلکه ممکن است تخصیص این نقش ها به صورت پویا باشد. مثلا ممکن است در یک جلسه پس از پخش یک رسانه توسط شخص A، شخص B بخواهد نقش میزبانی را بر عهده گرفته و رسانه دیگری را پخش نماید. پس مکانیزمی نیاز است که نقش های مختلف را میان اعضا به درستی مدیریت نماید.

۶. هماهنگی صوت: اگر صوت را به صورت یک موج در نظر بگیریم، وقتی چند منبع موج در مکان های متفاوت اقدام به پخش یک موج صوتی یکسان می نمایند، گوش انسان در یک آستانه خاصی از پارامتر ها و ناهماهنگی ها قادر به لذت بردن از صوت خواهد بود. این حدود آستانه برای پارامتر های مختلف موج برقرار است. سامانه بایستی این حدود آستانه را رعایت نماید. این پارامتر ها را می توان موارد زیر در نظر گرفت:

a. تاخیر زمانی موج ها (یا تاخیر فاز موج): تاخیر زمانی دو موج صوتی دریافتی توسط یک شنونده که می تواند متاثر از ناهماهنگی های زیر باشد:

i. ناهماهنگی زمانی پخش کننده ها به علت دقت زمان بند: هر روش یکسان سازی زمان دستگاه ها یک دقت عملیاتی دارد که متناسب با آن موجب افزایش ناهماهنگی در سامانه می شود.

ii. ناهماهنگی ایجاد شده در فرایند پخش محتوا: متاثر از متفاوت بودن سرعت سخت افزار ها و یا سیستم عامل های متنوع موجود در پخش همگانی محتوا

iii. ناهماهنگی ایجاد شده در اثر تفاوت فاصله منابع صوتی با گوش شنونده: این ناهماهنگی ناشی از سرعت انشار صوت در محیط است. در ملاحظات مربوط به این ناهماهنگی باید پارامتر شدت صوت را نیز در نظر گرفت، مثلاً ممکن است به علت فاصله زیاد تاخیر صوتی دو منبع خیلی زیاد شود ولی این مسئله به علت تضعیف صوت در این فاصله فاقد اهمیت باشد و نیاز نباشد برای آن چاره ای اندیشید.

b. تفاوت دامنه موج ها (یا بلندی صدا): یک سامانه مناسب بایستی شدت صوت تولید شده توسط دستگاه ها را نیز در اختیار خود داشته باشد یا دست کم برای آن مقدار پیشنهادی ارائه دهد.

c. جهت موج: از آنجایی که انسان دارای دو گوش می باشد، می تواند عمق اصوات و محل قرار گیری منابع صوتی را تشخیص دهد. اهمیت این پارامتر به میزانی است که نحوه چینش ساز ها در کنسرت های موسیقی از اصول اولیه کیفی کنسرت ها است. یک سامانه مناسب باید بتواند مشابه این حس شنیداری را در کاربر ایجاد نماید.

یک سامانه ایده آل بایستی مقدار آستانه را برای این پارامتر ها در نظر داشته باشد و پارامتر ها را پایین تر از حد آستانه نگه دارد.

۷. هماهنگی های دیگر (همه نوع هماهنگی به غیر از هماهنگی صوت): عملیات های هماهنگی متنوعی را می توان برای این سامانه متصور شد که تنها به هماهنگی های صوت محدود نمی شوند، مانند هماهنگی های کنترل صوت (اینکه حجم صدای دستگاه ها یکی باشد، تعویض صوت و ...)، هماهنگی های انتخاب نقش (مثلاً اینکه در صورتی که دو نفر قصد میزبان بودن داشته باشند نقش در نهایت به چه کسی می رسد). هماهنگی های مربوط به ملحق شدن و ترک دستگاه ها در حین پخش رسانه و هماهنگی های دیگر. سامانه بایستی این دسته از هماهنگی ها را نیز مدیریت کند. (علت جداسازی هماهنگی های پخش از هماهنگی های دیگر این است که چالش های موجود در هماهنگی های پخش به علت وابستگی به ذات موجی صوت با یکدیگر مشابهت هایی خواهند داشت که قرار دادن آن ها در یک دسته جداگانه می تواند طراحی کلی سامانه را بهبود ببخشد. علت مهم تر این جداسازی می تواند این باشد که هماهنگی های صوتی دارای سخت گیری بیشتری می باشند و احتمالاً چالش های بیشتری جهت نیل به این نیازمندی خواهیم داشت).

۸. پایداری و قابلیت اطمینان: سامانه باید قابل اطمینان بوده و پایدار باشد. در سامانه نباید هیچ نقطه شکست واحد داشته باشیم و کاربران در حین استفاده نباید دچار اختلال های این چنینی شوند. این نیازمندی به طور مستقیم بر هدف شفافیت تاثیر می گذارد.

۹. ارتباط شبکه ای میان دستگاه ها: این ارتباط بایستی به طور دقیق تعریف شود، یک چارچوب قابل اتکا برای این نیازمندی می تواند لایه های شبکه در استاندارد OSI باشد. مثلاً می توان تقریباً به طور قطع گفت که استفاده از پروتکل های لایه پردازش سیگنال یا لایه ۱ بدون ایجاد نوآوری خاصی خواهد بود چراکه این لایه بسیار ابتدایی بوده و درضمن احتمالاً کاربرد هایی همانند این سامانه در آن پیش بینی شده اند و تنها کفایت میان پروتکل ها و تجهیزات مختلف جستجو کرده و از میان آن ها انتخاب های مناسبی انجام داد. چنین قطعیتی در لایه های بالاتر تعمیم نمی یابد و در صورت لزوم بایستی پروتکل و معماری خاص منظوره تر برای این کاربرد طراحی کرد. در این سامانه، ارتباط شبکه ای میان اجزا برای دو کاربرد قابل تصور است، اولاً به عنوان یک بستر جهت انتقال اطلاعات رسانه (موسیقی یا فیلم) و دوماً به عنوان یک بستر جهت انتقال پیام های کنترلی مورد استفاده ی قسمت های مختلف سامانه (مانند پیام های کنترلی هماهنگ کننده، پیام های کنترلی کنترل رسانه (پخش و توقف موسیقی)، نکته ی حائز اهمیت در این مسئله این است که این دو بستر لزوماً یکی نیستند و در صورت نیاز می توان این دو را مجزا در نظر گرفت. (مثلاً بستر شبکه ی وای فای برای انتقال رسانه و بستر بلوتوث برای پیام های کنترلی)

۱۰. قابلیت مقیاس پذیری در ابعاد مختلف: مقیاس پذیری را در ابعاد مختلف می توان برای این سامانه متصور شد که شامل موارد زیر است:

a. مقیاس پذیری در بعد مسافت: بتوان بدون اعمال تغییر اساسی در طراحی، در محدوده ی جغرافیایی بزرگتری سامانه را اجرا کرد.

b. مقیاس پذیری در بعد تعداد کاربر: بتوان بدون اعمال تغییرات اساسی در طراحی، تعداد کاربران را به راحتی افزایش داد.

c. مقیاس پذیری در بعد کیفیت موسیقی و یا فیلم در حال پخش: بتوان بدون اعمال تغییرات اساسی در طراحی، حجم و کیفیت داده ی رسانه در حال پخش را افزایش داد. (پشتیبانی از فرمت های صوتی همچون FLAC و کیفیت های ویدئویی HD)

چالش‌ها:

چالش‌ها موارد اجتناب ناپذیری هستند که وجود آن‌ها را در فرضیات به عنوان عوامل منفی باید بپذیریم و در سامانه سعی کنیم برای آن‌ها چاره بیاندیشیم. با این تعریف مجموعه چالش‌های زیر را می‌توان برای سامانه متصور شد:

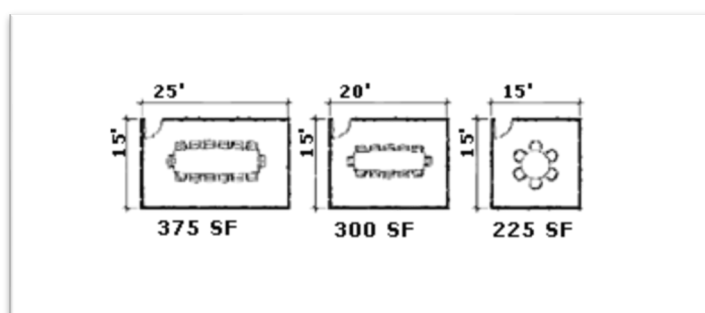
۱. موارد اجتناب ناپذیر و نامطلوب شبکه‌ای مانند تاخیر انتشار و تاخیر انتقال، جیتر و گم شدن بسته‌ها (می‌توان از تکنیک‌های فشرده سازی جهت کاهش تاخیر انتقال و پهنای باند مصرفی استفاده کرد. همچنین برای مقابله با این جیتر و گم شدن بسته‌ها می‌توان اطلاعات صوت را قبل از پخش به صورت پیشاپیش در دستگاه محلی بافر کرد).
۲. قدرت پردازش ناهمگون گوشی‌ها
۳. سرعت و حجم حافظه‌ی متفاوت گوشی‌ها
۴. تفاوت توان خروجی صوتی دستگاه‌ها: این پارامتر ممکن است در مطلوب بودن خروجی صوت نهایی تاثیرگذار باشد، مثلاً یک دستگاه با توان خروجی صوتی بالا ممکن است برای هماهنگ شدن با دیگر دستگاه‌ها، اقدام به کاهش حجم صدای خروجی خود نماید.
۵. تفاوت حجم و ذخیره باتری دستگاه‌ها: برخی دستگاه‌ها به علت فرسوده بودن باتری و یا عوامل دیگر ممکن است در مصرف توان دچار محدودیت‌هایی باشند. از آنجایی که تصمیمات طراحی (مانند پروتکل فشرده سازی، فرمت پخش، استفاده از شبکه‌های مبتنی بر LTE و ...) ممکن است تاثیر جدی بر میزان مصرف توان داشته باشند، ممکن است برای افزایش طیف انواع دستگاه‌ها، میل به استفاده از روش‌های کم مصرف تر داشته باشیم.
۶. یکسان نبودن ساعت دستگاه‌ها: اگر بخواهیم از برچسب‌های زمانی برای پخش نمونه‌های صوتی استفاده کنیم، بایستی این نکته را در نظر بگیریم که این برچسب‌ها به صورت پیش فرض از ساعت‌های ناهماهنگ دستگاه‌ها مشتق شده است.
۷. شبکه انتقال داده و پیام‌های کنترلی می‌تواند ناهمگون باشد. مثلاً ممکن است گروهی از دستگاه‌ها توسط وای‌فای و گروه دیگری از دستگاه‌ها توسط بلوتوث با یکدیگر در ارتباط باشند، تفاوت پروتکل‌ها و ویژگی‌های فیزیکی این محیط‌ها موجب افزایش درجه سختی راهکار خواهد شد.
۸. وجود عناصر بدرفتار در محیط مانند نقض کنندگان حریم خصوصی (مثلاً حریم خصوصی الگوی مصرف و حریم خصوصی مکانی)، تهدید کنندگان امنیت (مثلاً حمله DDOS) و استراق سمع کنندگان (eavesdroppers) در محیط

قسمت اول- اتاق جلسه

تطبیق اهداف، نیازمندی ها و چالش ها

تطبیق اهداف

۱. تعداد کاربران مطلوب سامانه بازه ی ۱ الی ۱۰ کاربر می باشد.
۲. محیط برابر با اتاق جلسه است. یک اتاق جلسه رایج برای جلسات تا حداکثر ۱۰ نفر دارای ابعاد ۳متر در ۷.۶متر خواهد بود. [2]



شکل (۱) ابعاد و ساختار رایج برای اتاق جلسه [2]

تطبیق پیش نیاز ها

۱. از آنجایی که محیط اتاق جلسه است، در انتخاب شبکه ی ارتباطی میان دستگاه ها اختیار بیشتری داریم. یک سویچ لایه ۳ (و یا حتی لایه ۲) در یک اتاق جلسه، با استفاده از Wifi می تواند ارتباط میان حداکثر ۱۰ دستگاه را به راحتی برقرار نماید. علاوه بر اتصال Wifi می توان از روش های دیگر مانند Bluetooth نیز استفاده کرد که با توجه به پهنای باند تمایل به استفاده از Wifi بیشتر است.
۲. از آنجایی که کاربران استفاده کننده از این سامانه همگی در یک اتاق جلسه حضور دارند و تعداد کاربران کم است، احتمالاً فرایند تخصیص نقش در این کاربرد ساده تر خواهد بود چراکه می توان این فرایند را به خود کاربران سپرد تا با یکدیگر به توافق برسند.
۳. هماهنگی های صوت در این سامانه به اشکال زیر متأثر از ویژگی های کاربرد خواهند بود:

a. پارامتر تاخیر زمانی موج ها: خود متشکل از تاخیر های زیر است:

i. ناهماهنگی های ناشی از دقت زمان بند: به علت تعداد کاربر کم و فضای محیطی کنترل شده، نیازی به استفاده از چنین منبع زمانبند نخواهیم داشت و این خود موجب افزایش دقت زمانبندی خواهد بود.

ii. ناهماهنگی های ناشی از تفاوت فاصله: در این کاربرد قابل صرف نظر کردن خواهد بود.

b. پارامتر تفاوت دامنه موج ها: به علت تعداد پایین دستگاه ها، ناهماهنگی شدت صوت محسوس تر خواهند بود پس باید برای آن چاره ای اندیشید.

طراحی اول

پیش فرض های اختصاصی

مجموعه ای از پیش فرض ها که برای این طراحی وجود دارند که شامل موارد زیر هستند:

- یک شبکه زیرساختی بیسیم وجود دارد که دستگاه ها می توانند به آن متصل شده و آدرس IP محلی دریافت نمایند. این شبکه توسط یک مسیریاب ایجاد شده و بر روی آن سرویس هایی همچون DHCP و ARP اجرا می شوند.
- تمامی دستگاه ها به شبکه متصل می باشند. (اینکه شبکه جهانی باشد یا محلی، بستگی به این دارد که سرویس های مبتنی بر شبکه که در ادامه معرفی خواهند شد در کدام شبکه در دسترس باشند).
- از معماری ارتباطی RESTful برای برقراری ارتباط اجزا استفاده می شود.

کشف سرویس و توزیع شناسه

مرحله اول: نحوه شناسه دهی

هر برنامه در ابتدای اجرا، بایستی اول شناسه یکتای سراسری (که از این پس به اختصار «شناسه» خواهیم گفت) خود را کشف نماید. برای این کار می توان از گزینه های زیر استفاده کرد:

- استفاده از آدرس MAC به عنوان شناسه
- استفاده از نسخه های مختلف استاندارد های تولید شناسه با عنوان UUID

استفاده از آدرس MAC به عنوان شناسه یکتای سراسری می تواند مشکلات زیادی ایجاد نماید. از جمله این مشکلات می توان به خطرات امنیتی ناشی از این طراحی اشاره کرد. [1] از آنجایی که هر دستگاه موجود در شبکه می تواند با استفاده از پروتکل ARP آدرس MAC میزبان های دیگر را بدست آورد، دستگاه های بدرفتار می توانند خود را به عنوان دستگاه های دیگر معرفی کنند. این کار موجب نقص شدید امنیتی در سامانه می شود.

استفاده از نسخه ی مناسب استاندارد تولید شناسه با عنوان UUID می تواند انتخاب بهتری باشد. این استاندارد مبتنی بر RFC 4122 بوده و دارای ورژن های مختلف برای کاربرد های مختلف می باشد که شامل موارد زیر هستند:

- نسخه ۱: مبتنی بر زمان
- نسخه ۲: نسخه امنیتی DCE، با POSIX UID های تعبیه شده
- نسخه ۳: نسخه ی مبتنی بر نام که از روش رمزگذاری درهم سازی MD5 استفاده می کند.

- نسخه ۴: نسخه تصادفی یا شبه تصادفی
- نسخه ۵: نسخه ی مبتنی بر نام که از روش رمزگذاری درهم سازی SHA-1 استفاده می کند.

با توجه به ویژگی های بالا، برای این سامانه استفاده از UUID4 مناسب خواهد بود.

پس دستگاه ها با استفاده از کتابخانه های مختلف ساخته شده برای ایجاد شناسه های مبتنی بر UUID4 در زبان های برنامه نویسی مختلف، در هربار اجرای سامانه اقدام به ساخت شناسه جدید می نمایند. البته برای جلوگیری از قطع اتصال در زمان بسته شدن برنامه، می توان شناسه ی تولید شده را برای مدتی در حافظه دستگاه ذخیره نمود و در زمان آغاز به کار برنامه در صورت موجود نبودن شناسه در حافظه، شناسه جدید ساخت.

مرحله دوم: کشف سرویس

دستگاهی که قصد میزبانی نشست را برعهده دارد اقدام به ساخت یک شناسه UUID4 دیگر برای خود می نماید. این شناسه دوم را شناسه میزبان نشست می نامیم. این شناسه جهت کاربرد کشف سرویس برای دستگاه های دیگر استفاده خواهد شد و شناسه اصلی میزبان نشست جهت کاربرد های آتی ذخیره خواهد شد. علت اینکه دستگاه میزبان نشست از شناسه اصلی خود برای این کار استفاده نمی کند این است که عمر شناسه اصلی هر دستگاه ممکن است بیشتر از عمر شناسه میزبانی نشست آن دستگاه باشد (مثلا یک دستگاه چندین بار اقدام به میزبانی نشست نماید). این جداسازی به امنیت سامانه و حریم خصوصی میزبان های نشست کمک می کند.

هر نسخه از سامانه به صورت از پیش تعبیه شده، آدرس حداقل دو nameserver سامانه را به صورت hard-coded دارد. سرویس nameservice ساده ای با تکنیک های تکرار یا replication از قبل ایجاد شده است و آدرس این ماشین ها به صورت آدرس IP و یا در صورت موجود بودن DNS محلی، آدرس دامنه در تمامی دستگاه های دیگر hard-code شده است. دستگاه ها از این سرویس دهنده ها برای یافتن نشست های محلی استفاده خواهند کرد.

هر کاربر باید در ابتدا در سامانه ثبت نام نماید و هرگاه قصد استفاده از سرویس nameserver را دارد باید در وضعیت «وارد شده» باشد. این احراز هویت جهت امور AAA می تواند استفاده شود.

میزبان نشست باید اطلاعات نشست ساخته شده خود را در nameserver ثبت کند. برای این منظور حداقل پارامتر های زیر نیاز است:

- شناسه نشست
- رمز نشست
- اطلاعات اتصال (نحوه دسترسی به میزبان نشست)

- نام کاربرپسند برای نشست (اختیاری)

که توسط میزبان نشست تامین می‌شود.

سپس دستگاه‌ها می‌توانند جهت افزوده شدن به نشست پارامتر شناسه نشست را در nameserver وارد کرده و پارامتر رمز نشست و اطلاعات اتصال به میزبان نشست را به صورت رمز شده از آن دریافت نمایند تا با استفاده از آن به میزبان نشست دسترسی پیدا کنند.

مرحله سوم: توزیع شناسه

پس از اینکه دستگاه‌ها به میزبان نشست دسترسی پیدا کردند، می‌توان فرایند توزیع کلید را انجام داد. این نقش می‌تواند توسط میزبان نشست به تنهایی انجام شود و بدین ترتیب همه ی شناسه‌ها در اختیار همه ی افراد موجود در نشست قرار می‌گیرد. این توزیع شناسه‌ها در آینده برای فرایند های تخصیص نقش استفاده خواهند شد.

علت اینکه وظایف توزیع شناسه را برعهده ی nameservice قرار ندادیم این است که از درجه متمرکز بودن سامانه بکاهیم و همچنین کار این سرویس را مختصر و سبک نماییم.

هماهنگی های صوت

انتخاب هماهنگ کننده صوت

یکی از دستگاه‌ها باید نقش هماهنگ کننده صوتی را ایفا کند. نحوه انتخاب این دستگاه می‌تواند به شیوه های مختلف باشد ولی باید درنظر داشت که انتخاب این دستگاه بایستی بدون تصمیم گیری مرکزی باشد و به صورت توزیع شده مابین دستگاه‌ها تصمیم گیری انجام شود. اگر این تصمیم گیری به صورتی غیر از توزیع شده بودن انجام پذیرد، تک نقطه شکست خواهیم داشت. همچنین تخصیص این نقش از نظر کاربر باید شفاف باشد. پیش از انجام تصمیم گیری فرض های زیر را انجام می‌دهیم:

- تمامی گوشی‌ها دارای شناسه می‌باشند.
- تمامی گوشی‌ها شناسه ی تمامی گوشی‌های موجود در نشست را می‌دانند. (ولی از روشن یا خاموش بودن آن‌ها اطلاعی ندارند).

به صورت پیش فرض بهترین دستگاه برای قبول نقش هماهنگ کننده صوت همان میزبان نشست می‌باشد. چراکه این دستگاه تمامی دستگاه‌های دیگر را می‌شناسد و دستگاه‌های دیگر نیز این دستگاه را می‌شناسند. اما ممکن است عواملی همچون منابع ناکافی (باتری، پردازش، ذخیره سازی و شبکه) و یا عوامل دیگر موجب شوند که بهتر باشد نقش هماهنگ کننده صوتی به صورت هوشمندانه تری تخصیص شود (میزبان نشست

صرفاً دستگاهی بود که اقدام به ساخت نشست پخش همگانی صوت نموده بود). برای این منظور می‌توان از یک الگوریتم مناسب جهت انتخاب بهترین دستگاه برای قبول نقش هماهنگ کننده ی صوت استفاده کرد. سپس اطلاعات هماهنگ کننده ی صوت جدید را در میزبان نشست قرار داد. بدین صورت دستگاه ها می‌تواند با واسطه میزبان نشست به هماهنگ کننده ی صوت دسترسی داشته باشند.

جهت بهبود عملکرد سامانه می‌توان مکانیزم شناسایی هماهنگ کننده ی صوتی برای دستگاه های تازه وارد را بهبود داد. به صورت پیش فرض این فرایند به این صورت انجام می‌شود که دستگاه تازه وارد اطلاعات ارتباط با میزبان نشست را از NameService دریافت می‌کند و پس از برقراری ارتباط با آن، اقدام به پرسیدن اطلاعات ارتباط با هماهنگ کننده ی صوتی از میزبان نشست می‌نماید. می‌توان جهت بهبود این فرایند اینگونه عمل کرد که میزبان نشست در صورت تمایل اطلاعات هماهنگ کننده ی صوت را در NameService ثبت کند تا دستگاه های جدید بدون واسطه بتوانند به آن متصل شوند.

این رویکرد علاوه بر اینکه موجب افزایش سرعت اضافه شدن دستگاه های جدید می‌شود، بار وارد بر میزبان نشست را به NameService منتقل می‌کند و از آنجایی که سرویس NameService در چند ماشین Replicate شده است، قابلیت اطمینان سامانه را نیز بهبود می‌دهد.

الگوریتم مناسب جهت انتخاب بهترین دستگاه برای قبول نقش هماهنگ کننده ی صوت:

در رابطه با این الگوریتم می‌توان موارد زیر را روشن نمود:

- این الگوریتم پارامتر های ورودی متنوعی دارد. از جمله اطلاعات منابع سخت افزاری و نرم افزاری تمامی دستگاه های کاندید، سیستم عامل آن‌ها، توپولوژی تقریبی شبکه و ...
- این الگوریتم به صورت مرکزی (و نه توزیع شده) تنها در میزبان نشست اجرا خواهد شد.
- دستگاه های دیگر میزبان نشست را به عنوان هماهنگ کننده صوت در نظر میگیرند مگر اینکه میزبان نشست دستگاه دیگری را برای این نقش معرفی کند.
- این الگوریتم با توجه به پارامتر ها و محدودیت ها، تابع هدف بهینه کردن تجربه شنیداری صوت را خواهد داشت.

وظایف هماهنگ کننده ی صوت

وظیفه اصلی هماهنگ کننده ی صوت اجرای الگوریتم NTP [6] جهت یکسان سازی زمان دستگاه های موجود در نشست پخش همگانی می‌باشد. این ساعت همسان شده در فرایند پخش محتوا مورد استفاده خواهد بود.

هماهنگی های دیگر (کنترل محتوا)

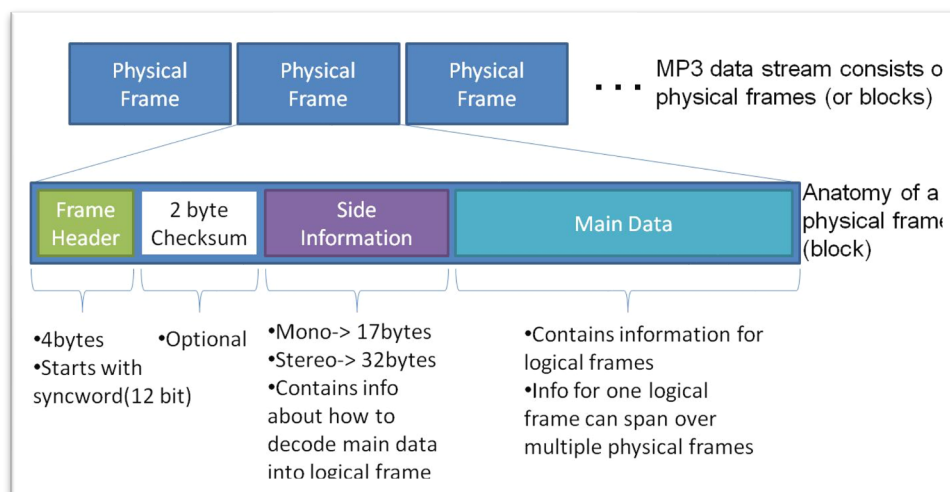
انتخاب و تغییر هماهنگ کننده اعمال دیگر

نحوه انتخاب و تغییر این نقش نیز می‌تواند همانند انتخاب و تغییر نقش هماهنگ کننده ی صوت باشد. بدین صورت که این نقش به صورت پیش فرض برعهده میزبان نشست باشد تا در صورت نیاز میزبان نشست دستگاه بهتری را برای این کار معرفی کند. تنها تفاوت انتخاب این نقش با نقش هماهنگ کننده ی صوت این است که این نقش نیازمند الگوریتم اختصاصی مناسب جهت انتخاب نقش را خواهد داشت. چراکه پارامترهای تصمیم گیری و محدودیت ها و تابع هدف آن متفاوت است. این الگوریتم به دنبال بیشینه کردن تجربه شنیداری نخواهد بود چراکه خروجی این الگوریتم تاثیر نسبی کمتری بر روی تجربه شنیداری خواهد داشت (مثلا مطمئنا بازه تحمل خطای زمانی افزایش حجم صدای دستگاه ها، گسترده تر از بازه خطای زمانی فاز موج شنیداری خواهد بود).

اجرا کننده ی محتوا

اجرای فرمت MP3

این فرمت از صوت از واحد هایی به نام frame تشکیل شده است که هر کدام از آن ها قسمتی از اطلاعات صوت را حمل می کنند. هر frame از قسمت های متعددی تشکیل شده است. شکل زیر نمایانگر قسمت های مختلف MP3 frame است.



شکل ۲) فرمت MP3 frame به صورت شماتیک [4]

هماهنگ سازی در پخش این فرمت با استفاده از تنظیم اطلاعات موجود در قسمت بدنه frame و تنظیم پارامترهای متناسب با آن در سرآیند و سپس ارسال هماهنگ frame ها از منابع متعدد قابل دستیابی است. سرآیند این بسته ها مشخص کننده ی پارامترهای اندازه ی فریم، نرخ بیت و نرخ نمونه برداری است که با تنظیم این پارامترها متناسب با اندازه ی اطلاعات موجود در بدنه به همراه زمان بندی ارسال مناسب، می‌توان به پخش هماهنگ موسیقی از دستگاه های مختلف دست یافت. بزرگ ترین چالش در این کار، یکسان کردن

ساعت دستگاه های مختلف است که این امر بوسیله ی هماهنگ کننده ی صوت و استفاده از پروتکل NTP محقق می شود.

توزیع کردن محتوا

از آنجایی که برای پخش همزمان صوت علاوه بر اطلاعات زمانی هماهنگ نیازمند داده های مشترک نیز هستیم، بایستی یک ساز و کار مناسب جهت توزیع محتوا در سامانه تدارک ببینیم. نکته ی قابل توجه در طراحی این ساز و کار این است که می توان محتوا را بسیار زودتر از برچسب زمانی پخش آن در بین دستگاه ها توزیع کرده و در آنجا عمل بافرینگ انجام داد. بافر کردن داده ها پیش از پخش تاثیرات مثبت و منفی زیر را بر سامانه دارد:

- افزایش تحمل پذیری سامانه در مقابل چالش های شبکه ای مانند تاخیر و جیتر
- اختیار و کنترل بیشتر در فرایند های مهندسی ترافیک: زیاد بودن تاریخ انقضای قطعه های اطلاعاتی، فرصت بیشتری در اختیار طراح سامانه قرار می دهد که این قطعه ها را جهت نیل به اهداف دیگر سامانه به طور دلخواه هدایت کند. مثلاً حالتی را می توان متصور شد که توزیع محتوا را به طور سلسله مراتبی انجام دهیم. زیاد بودن تاریخ انقضای هر قطعه این امر را راحت تر می کند.
- مصرف حافظه بیشتر در دستگاه ها
- مصرف پهنای باند بیشتر در دستگاه ها: سناریویی را در نظر می گیریم که قسمتی از صوت در بافر قرار گرفته ولی به دلایلی چون عوض شدن فایل صوتی و یا توقف پخش، از بافر بدون پخش شدن حذف می شود.

نکته ی قابل ذکر در اینجا این است که تصمیم گیری بر سر انجام یا عدم انجام بافرینگ نیست، چراکه بافر کردن اجتناب ناپذیر است؛ بلکه چالش اصلی رسیدن به مصالحه ی مناسب میان خوبی ها و بدی های بافر کردن با استفاده از پارامتر تاریخ انقضا (اینکه چقدر زودتر محتوا را پخش می کنیم) است. در یک سمت مصالحه حالتی قرار دارد که سعی کنیم کمترین میزان بافرینگ را داشته باشیم (با استفاده از پارامتر های شبکه و مدل کردن توزیع آن ها و محاسبه احتمال دیر رسیدن) و در سمت دیگر مصالحه، می توان کل فایل صوتی را در ابتدا بین تمامی دستگاه ها پخش نمود. واضح است که هر دو انتخاب نامطلوب هستند. در انتقال کل فایل رفتار شبکه ناپایدار می شود و ممکن است در لحظه های ابتدای پخش موسیقی ازدحام رخ دهد. همچنین در این قسمت از سامانه بسته به نیازمندی می توان مکانیزم های توازن بار پیاده سازی کرد تا فشار عملیاتی بر روی چند دستگاه تقسیم شوند.

ارزیابی طراحی اول

این طراحی ایرادات زیر را دارد:

۱. فرض شده است که یک شبکه زیرساختی آماده وجود دارد که در حالت های زیادی این فرض درست نیست و جامعیت سامانه پایین است.

۲. از آنجایی که شناسایی میزبان نشست در این سامانه نیازمند شناسایی و استفاده از یک name-service است این مشکل وجود دارد که بایستی یک name-service مناسب با قابلیت های توزیع شدگی و قابلیت تحمل خرابی بالا تعبیه شود که هزینه نگهداری سامانه را بالا می برد.

۳. از زمانی که یک کاربر تصمیم به اضافه شدن به نشست می گیرد تا شروع پخش آن کاربر در پخش همگانی، عملیات و فرایندهای زیادی بایستی انجام شوند (اشتراک رمز نشست، برقراری ارتباط با name-server و انجام نیازمندی های احراز هویت و AAA، برقراری ارتباط با میزبان نشست و دریافت اطلاعات هماهنگ کننده ی صوتی و ...) بایستی در نظر داشت که این فرایندها در صورت زمان بر بودن تجربه کاربر را در استفاده از این سامانه نامطلوب می نمایند.

طراحی دوم

پیش فرض های اختصاصی

در این طراحی فرض می شود در اتاق جلسه یک دستگاه قابلیت wifi hotspot گوشی همراه خود را روشن کرده و به عنوان نقطه دسترسی به دستگاه های دیگر موجود در اتاق جلسه سرویس ارائه می نماید.

کشف سرویس و توزیع شناسه

اولین قدم در راستای فرایند کشف سرویس در این طراحی این است که دستگاه ها بتوانند به شبکه ی wifi به اشتراک گذاشته شده متصل شوند. این فرایند را می توان هم در سامانه تعبیه کرد و هم از قابلیت های ارائه شده توسط سیستم عامل (در خارج سامانه) استفاده نمود. جهت سادگی فرض می شود در این طراحی فرایند های روشن نمودن wifi hotspot و نیز یافتن و ملحق شدن به این شبکه توسط کاربر به صورت دستی انجام می شود. در سامانه تنها فرایند های کنترلی مربوط به عملیات بالا انجام گرفته و دستورالعمل انجام فرایند ها به صورت متنی به کاربر نمایش داده می شود.

دستگاه ها جهت کشف سرویس wifi hotspot ارائه شده می توانند از قابلیت لینک اشتراک شبکه که توسط سیستم عامل گوشی همراه ارائه می شود استفاده نمایند. این لینک در قالب QR code به نمایش درآمده و باقی دستگاه ها می توانند با دوربین تلفن همراه خود این کد را اسکن کرده و وارد شبکه ی پخش همگانی شوند.

در این طراحی فرایند احراز هویت با استفاده از شبکه های wifi hotspot رمز دار مدیریت می شود. بدین ترتیب دیگر نیازی به طراحی مکانیزم دیگری جهت انجام احراز هویت نخواهیم داشت. مشکلی که این روش طراحی ایجاد می نماید این است که امنیت و حریم خصوصی دستگاه میزبان hotspot را دچار مخاطره می نماید. چراکه این دستگاه باید رمز ورود به hotspot خود را در اختیار سایرین قرار دهد و اگر پس از پایان نشست پخش همگانی اقدام به تعویض رمز مذکور ننماید، دستگاه های دیگر همچنان این مجوز را در اختیار خواهند داشت که به این شبکه وارد شوند.

جهت رسیدگی به این مشکل ایجاد شده، یک رویکرد این است که سامانه خود اقدام به تنظیم رمز عبور برای هر نشست نموده و پس از پایان نشست رمز عبور را به حالت عادی بازگرداند.

پس از آنکه یک دستگاه وارد شبکه شد، UUID خود که مطابق با طراحی اول تولید شده را در اختیار میزبان hotspot قرار می دهد و لیست شناسه های دستگاه های دیگر موجود در پخش همگانی را از میزبان hotspot دریافت می نماید. میزبان hotspot وظیفه جمع آوری و توزیع این شناسه ها را بر عهده دارد.

در این توپولوژی شبکه، تمامی دستگاه ها فقط با میزبان hotspot ارتباط می گیرند و این دستگاه همانند یک hub عمل می کند. هر پیامی مابین دو دستگاه باید توسط میزبان hotspot هدایت شود.

هماهنگی ها

در این طراحی فرض می‌شود که تمامی هماهنگی ها (از جمله هماهنگی های صوتی و هماهنگی های کنترل محتوا و ...) در یک دستگاه انجام می‌شود و همانند طراحی قبل آن ها را از یکدیگر جدا نمی‌کنیم. این تصمیم طراحی دلایل متعددی می‌تواند داشته باشد. به عنوان مثال:

- اگر بخواهیم نقش هماهنگی ها را همانند طراحی قبل در دستگاه های متعدد توزیع کنیم، تمامی پیام های ارسال شده توسط این دستگاه ها باید از میزبان hotspot عبور کرده و به دیگر دستگاه ها ارسال شوند. پس جداسازی این فرایند ها در بهبود عملکرد سامانه (مخصوصا در پارامتر تاخیر) چندانی نخواهد داشت.
- به دلیل تعداد پایین دستگاه های موجود در پخش همگانی، سر بار انجام هماهنگی های متنوع (از جمله هماهنگی های صوتی و هماهنگی های کنترل رسانه و ...) قابل توجه نخواهد بود و میزبان hotspot می‌تواند آن ها را به عهده بگیرد.

توزیع محتوا

اطلاعات محتوای صوتی در این طراحی می‌تواند به راحتی توسط همه پخش مابین تمامی دستگاه های موجود توزیع شود؛ ولی باید توجه داشت که برای این کار بایستی میزبان hotspot فایل صوتی را در اختیار داشته باشد. پس در صورتی که نقش میزبان پخش به کسی غیر از میزبان hotspot داده شده باشد، بایستی در ابتدا و پیش از شروع اجرای پخش همگانی صوت، کل فایل صوتی به میزبان hotspot منتقل شود.

ارزیابی طراحی دوم

این طراحی دارای اشکالات زیر می‌باشد:

۱. این طراحی حول موجودیت نشست میزبان و یا میزبان hotspot شکل گرفته است. نقش های زیادی به این دستگاه داده شده (نقش هماهنگ کننده صوت، هماهنگ کننده کنترل محتوا، توزیع محتوا و ...) که این موجب افزایش بار محاسباتی و ذخیره سازی در این دستگاه خواهد شد. بنابراین حتی اگر این دستگاه کشش سخت افزاری ایفای نقش در این شرایط را داشته باشد دچار افزایش شدید مصرف باتری خواهد شد.
۲. در این طراحی در صورت از دسترس خارج شدن میزبان hotspot، کل نشست دچار فروپاشی شده و همه ی فرایند ها بایستی با میزبان جدید از سر گرفته شوند. در واقع میزبان hotspot نقطه شکست واحد این سامانه است.
۳. از آنجایی که باید رمز hotspot میزبان نشست در اختیار دیگر کاربران قرار گیرد نقص امنیتی داریم. جهت بهبود این نقص می‌توان از رمز عبور موقت استفاده کرد ولی این مکانیزم ممکن است در تمامی

سیستم عامل ها به برنامه ها داده نشود (مثلا سیستم عامل iOS در دسترسی های این چنینی سختگیرانه عمل می نماید) و یا در صورت وجود داشتن این امکان نیازمند افزایش دسترسی برنامه باشد (مانند دسترسی root در اندروید) که خود نقص امنیتی شدید تری را به دنبال خواهد داشت. ۴. این طراحی قابلیت مقیاس پذیری در بعد تعداد کاربر و بعد مسافت را ندارد و یک دستگاه امروزی

توانایی پشتیبانی از hotspot حداکثر برای ۱۰ دستگاه دیگر را دارد. [7]

۵. در صورت استفاده از رمز عبور موقت، تضمین بازنشانی رمز عبور کاربر در پایان نشست پخش همگانی دشوار خواهد بود. حالت هایی را می توان در نظر گرفت که نشست به صورت صریح و قاعده مند پایان نمی یابد و عواملی همچون خاموش شدن میزبان و یا kill شدن سامانه نشست را پایان می دهند. در این حالت ها ممکن است رمز عبور تنظیم شده توسط کاربر بازنشانی نشده و در استفاده های آتی کاربر از قابلیت hotspot در گوشی خود دچار مشکل شود.

مقایسه ی طراحی ها

طراحی اول با دیدگاه افزایش مقیاس پذیری ارائه شده است و از این رو فرضیات کلی بیشتری در رابطه با شبکه ی زیرساختی خود دارد که باید برقرار باشند. این فرضیات با وجود اینکه مقیاس پذیری سامانه را افزایش می دهند ولی جامعیت کاربرد را کاهش خواهند داد.

در طراحی دوم سعی شده است که کمترین پیش فرض را برای اجرا در نظر بگیریم و سعی کنیم سادگی برقراری نشست حفظ شود. در عوض این طراحی برای کاربرد های بیشتر از ۱۰ نفر غیرقابل استفاده خواهد بود. در واقع این دو طراحی دو جواب متفاوت در مصالحه ی میان پارامتر های سادگی و گسترش پذیری هستند.

قسمت دوم- تعمیم به استادیوم

از آنجایی که در طراحی اول ارائه شده در قسمت پیشین تمرکز بر روی ویژگی مقیاس پذیری سامانه بود، جهت تعمیم مسئله به کاربرد ورزشگاه آزادی مناسب خواهد بود. در ادامه به توسعه های مورد نیاز در طراحی ۱ از قسمت پیشین پروژه جهت پاسخ گویی به نیازمندی ۱۰۰ هزار نفر (که در مساحت ورزشگاه آزادی پخش شده اند) پرداخته خواهد شد. درواقع در ادامه تنها به ذکر توسعه های مورد نیاز بر روی طراحی ۱ پرداخته خواهد شد و بقیه موارد که ذکر نشده اند تغییر ماهوی چندانی نخواهند داشت.

در این طراحی فرض می شود یک شبکه زیرساختی بی سیم با استفاده از مسیریاب های لایه ۲ و ۳ ایجاد شده است که دستگاه ها می توانند در این شبکه آدرس IP محلی دریافت نمایند. با توجه به تعداد دستگاه ها، نیازمند حداقل ۱۷ بیت برای آدرس دهی host ها یا همان دستگاه ها خواهیم بود. پس آدرس دهی باید دارای ماسک شبکه ی ۱۵ باشد. مدیریت این تعداد دستگاه در یک شبکه موجب کاهش شدید بازدهی می شود پس بایستی این تعداد نفرات را در زیرشبکه های کوچک تر قرار داد. سپس ارتباط میان این زیر شبکه ها را با استفاده از سویچ های لایه ۳ و یا مسیریاب ها برقرار نمود.

در صورت استفاده از سویچ های لایه ۳ و یا مسیریاب ها، می توان برخی از سرویس های مورد نیاز شبکه مانند ARP و DHCP و DNS را بر روی این سویچ ها اجرا کرد. برای این منظور می توان به جای استفاده از سخت افزار های خاص منظوره، از تکنیک مجازی سازی عناصر شبکه ای یا NFV استفاده نمود. [8] مثلاً می توان در یک سخت افزار با استفاده از یک hypervisor سطح ۱ چندین ماشین مجازی ایجاد نمود و مثلاً یکی از این ماشین های مجازی وظیفه مسیریابی و یکی وظیفه اجرای پروتکل ARP را ایفا نماید.

در فرایندی که هر دستگاه اقدام به ساخت UUID برای خود می نماید، در صورت استفاده از UUID4 از نظر تعداد دستگاه ها مشکل نخواهیم داشت چراکه این ورژن از UUID از ۱۲۲ بیت تصادفی استفاده می کند. [3] این فضای بالای تولید شناسه علاوه بر اینکه تعداد شناسه ۱۰۰ هزار عدد را پوشش می دهد احتمال برخورد با شناسه های یکسان را نیز بسیار پایین و قابل صرف نظر کردن می کند.

نقش میزبان نشست در این طراحی به صورت توزیع شده پیاده سازی خواهد شد. در صورتی که این نقش به صورت متمرکز باشد در زمان شروع نشست حجم بالایی از درخواست ها به دستگاه دارای این نقش وارد شده و ایجاد مشکل خواهد کرد. از طرفی پیاده سازی توزیع شده ی این نقش پیچیدگی بالایی نخواهد داشت. برای پیاده سازی توزیع شده ی نقش میزبان نشست می توان از سخت افزار های تعبیه شده برای مسیریاب ها استفاده نمود. دقت شود که این سخت افزار ها با استفاده از hypervisor سطح ۱ مجازی سازی شده و چندین ماشین مجازی بر روی آن ها قابل نصب است (که تنها یکی از آن ماشین های مجازی جهت امور

مسیریابی استفاده می‌شود) و می‌توان یک ماشین مجازی جهت اجرای سرویس مورد نیاز سامانه در آن تدارک دید. پس در این طراحی به تعداد زیرشبکه‌ها hypervisor ایجاد می‌نماییم و آن‌ها را در سرتاسر استادیوم توزیع می‌نماییم. از آنجایی که این دستگاه‌ها سیار نیستند می‌توان شبکه ارتباطی سیمی مثل Ethernet بین آن‌ها ایجاد نمود.

از آنجایی که این کاربرد وسیع و احتمالا تجاری است، توسعه سامانه به صورت چند application به صرفه خواهد بود. پس می‌توان یک سامانه مجزا و خاص منظوره جهت اجرای نقش میزبان نشست توسعه داد و آن را به صورت توزیع شده بر روی ماشین‌های مجازی مذکور اجرا کرد. در صورتی که نقش میزبان نشست در چند ماشین مجازی پیاده‌سازی شود، بهتر است از آدرس دامنه به جای آدرس IP جهت دسترسی به آن‌ها در فرایند کشف سرویس استفاده شود. چراکه با کمک سرویس دهنده‌های DNS محلی می‌توان دستگاه‌ها را به نزدیک‌ترین ماشین مجازی اجرا کننده‌ی سرویس مذکور هدایت کرد. البته باید توجه داشت که از کاربر انتظار می‌رود به صورت منطقی به قوی‌ترین (نزدیک‌ترین) زیر شبکه متصل شود.

در طراحی اول ذکر شد که کاربر ابتدا بایستی به name-server متصل شود تا اطلاعات نشست را بدست آورد. معادل این عمل در طراحی توسعه یافته این خواهد بود که کاربر به سرویس DNS شبکه خود متصل شده و اطلاعات میزبان نشست (که به صورت توزیع شده در زیرشبکه موجود است) را از آن دریافت می‌نماید. سپس به میزبان نشست متصل شده و اطلاعات نشست را دریافت می‌نماید.

علت اینکه در طراحی ۱ شناسه‌ی تمامی دستگاه‌ها را در تمامی دستگاه‌ها قرار دادیم این بود که بتوانیم تخصیص نقش پویا میان دستگاه‌ها داشته باشیم. در کاربرد استادیوم تخصیص نقش‌ها در ابعاد استادیوم بی‌معنی خواهد بود پس توزیع شناسه‌ها نیز در این ابعاد مطرح نیست. در این طراحی می‌توان تنها دستگاه‌های موجود در یک زیرشبکه را به یکدیگر شناساند (با استفاده از توزیع شناسه) و در نتیجه تخصیص نقش‌ها را در این سطح انجام داد. (جهت یادآوری تخصیص نقش برای انجام اموری همچون هماهنگی‌های صوتی، هماهنگی‌های دیگر و ... انجام می‌شد).

جهت توزیع شناسه در سطح زیرشبکه می‌توان در hypervisor یک ماشین مجازی در نظر گرفت و یا این نقش را در یکی از ماشین‌های مجازی (مسیریاب و یا میزبان نشست) ایفا نمود. در ادامه نقش‌های توصیف شده در پیش‌نیازها، به ازای هر زیرشبکه تخصیص داده خواهند شد. اگر از سخت افزار قوی‌تر در ماشین اجرا کننده‌ی hypervisor استفاده شود، می‌توان تمامی نقش‌ها را به این دستگاه‌ها داد ولی ممکن است بخواهیم از منابع دستگاه‌های کاربران جهت انجام امور استفاده نماییم که در این صورت الگوریتم‌های توصیف شده در طراحی ۱ قابل استفاده خواهند بود.

پس از تخصیص وظیفه‌ی هماهنگ کننده‌ی صوت، الگوریتم NTP بر روی این دستگاه‌ها اجرا خواهد شد تا زمان این دستگاه‌ها در زیر شبکه‌های مختلف با یکدیگر هماهنگ شود. سپس هر هماهنگ کننده اقدام به

یکسان سازی زمان در زیرشبکه خود می‌نماید. این فرایند نیز با استفاده از یک اجرای محلی NTP انجام می‌شود.

دقت زمان بندی در این طراحی متأثر از پارامتر های زیر خواهد بود:

- تاخیر ناشی از سرعت انتقال پیام های هماهنگی در شبکه یا transportation delay
- تاخیر ناشی از سرعت انتشار پیام های هماهنگی در محیط یا propagation delay
- تاخیر ناشی از تفاوت قدرت پردازش پیام های هماهنگی توسط دستگاه های ناهمگون موجود در محیط

از آنجایی که هماهنگی زمان در این طراحی در دو سطح انجام می‌شود، دقت آن پایین خواهد بود. ولی این نکته بایستی مورد توجه قرار بگیرد که دقیق بودن زمان بند دستگاه ها امری محلی است. درواقع نیازی نیست که دو دستگاه در دو سمت ورزشگاه با یکدیگر هماهنگ باشند چراکه اصلا صدای ایجاد شده در این فاصله با توجه به نویز محیط قابل شنیدن نخواهد بود. پس می‌توان حد آستانه ی خطای زمان بند دستگاه ها را با توجه به فاصله آن‌ها از یکدیگر و صدای محیط تعیین نمود که درجه آزادی سامانه را به شدت بهبود می‌دهد.

توزیع محتوا نیز در این طراحی می‌تواند سلسله مراتبی باشد. می‌توان کل یا قسمتی از محتوای صوتی را به ماشین های مجازی ارسال کرد و سپس از آنجا به دستگاه های موجود در زیرشبکه توزیع نمود. به طور کلی این دیدگاه سلسله مراتبی را می‌توان در قسمت های مختلف تعمیم داد.

قسمت سوم- محتوای تصویری

صفحه مجازی

هر دستگاه دارای یک صفحه نمایش می‌باشد و آن صفحه نمایش دارای طول و عرض مشخص است. نکته ای که در اینجا باید به آن توجه نمود این است که تنها طول و عرض صفحه نمایش جهت پخش محتوا در یک چینش دلخواه کافی نیست؛ چراکه رزولوشن یا تراکم پیکسلی صفحه نمایش نیز بر عملکرد پخش تاثیر می‌گذارد.

به صورت شهودی، کیفیت یک فیلم پخش شده توسط چند دستگاه، باید متناسب با رزولوشن کم کیفیت ترین دستگاه موجود در پخش همگانی باشد. این مسئله را می‌توان به صورتی دیگر بیان کرد. اگر یک دستگاه صفحه نمایش با رزولوشن بسیار بالا (به نسبت دستگاه های دیگر) داشته باشد مشکلی نخواهیم داشت چراکه می‌توانیم در برنامه پخش ویدئو، با زوم کردن مناسب رزولوشن ویدئوی مطلوب ایجاد کرد (با زوم کردن درواقع می‌گوییم که هر پیکسل ویدئو را بر روی بیش از پیکسل واقعی نمایش بده) ولی اگر یک دستگاه صفحه نمایش با رزولوشن بسیار پایین داشته باشیم نمی‌توانیم تعداد پیکسل های ویدئوی تخصیص یافته به آن دستگاه را در محدوده ی صفحه نمایش آن بگنجانیم.

برای این مشکل می‌توان مفاهیم زیر را تعریف نمود:

اندازه پیکسل شاخص: اگر ابعاد صفحه نمایش کم کیفیت ترین دستگاه موجود در پخش همگانی را تقسیم بر تعداد پیکسل های موجود در هر بعد آن نماییم، پارامتر ابعاد کم کیفیت ترین پیکسل را خواهیم داشت که آن را اندازه پیکسل شاخص می‌نامیم.

صفحه نمایش مجازی: این پارامتر برای هر دستگاه با ضرب نمودن تعداد پیکسل موجود در هر بعد آن در اندازه پیکسل شاخص بدست می‌آید.

طبق تعاریف بالا اندازه صفحه نمایش کم کیفیت ترین دستگاه برابر با اندازه صفحه نمایش حقیقی آن خواهد بود و بقیه دستگاه ها صفحه نمایش مجازی بزرگتر از صفحه نمایش واقعیشان خواهند داشت.

قاب فیلم: با تعاریف و نکات ذکر شده در بالا، ابعاد قاب فیلم را باید با استفاده از اندازه پیکسل شاخص تعیین نمود. اگر یک قطعه ویدئویی با یک فرمت دلخواه نیازمند $n*m$ پیکسل باشد و اندازه پیکسل شاخص $x*y$ باشد، اندازه ی قاب فیلم $nx*my$ خواهد بود.

چینش

در صورتی که بخواهیم از محتوای ویدئویی به جای محتوای صوتی استفاده نماییم، یکی از تفاوت ها این است که در ارائه ی دسته جمعی محتوای ویدئویی نحوه ی قرار گیری دستگاه ها یا «چینش» آن ها مهم خواهد بود. ابتدا سعی خواهد شد چینش دستگاه ها در فضا را مدل کنیم.

چینش را می‌توان مجموعه ی پارامتر های زیر در نظر گرفت:

- مختصات نسبی مرکز دستگاه ها (مرکز مستطیل)
- جهت گیری دستگاه در فضای سه بعدی (می‌توان آن را با سه پارامتر نمایش داد)

با پارامتر های فوق، می توان تمامی حالت های قابل تصور برای چینش هر تعداد دستگاه را در فضا تعریف کرد. ولی یک سری محدودیت و فرض داریم که در قالب شروط می توان به مسئله اضافه کرد:

شرط فیزیک:

- از آنجایی که یک نقطه از فضای سه بعدی تنها می تواند متعلق به یک دستگاه باشد، شرط شدنی بودن یک چینش این است که با اضافه شدن ابعاد دستگاه ها به چینش، هیچ نقطه از فضا به بیش از یک دستگاه اختصاص داده نشود.

شرط زیبایی بصری:

- یک چینش باید به صورتی باشد که زیبایی بصری ویدئو را حفظ نماید. از آنجایی که پارامتر زیبایی مبهم است، می توان هر «مدل سازی زیبایی» مناسبی را جهت رفع ابهام آن در سامانه در نظر گرفت.

می توان مجموعه ای از شروط و فرضیاتی را در مسئله در نظر گرفت که موجب سادگی ادامه فرایند می شوند. البته باید در نظر داشت که این فرضیات را می توان جهت دستیابی به اهداف دیگر کنار گذاشت.

شرط ساده شدن مسئله:

- مرکز تمامی دستگاه ها در یک صفحه قرار دارند (البته می توان این فرض را در نسخه های پیشرفته تر سامانه برای ساخت صفحه های مجازی دارای انحنا بهبود داد)
- جهت گیری های زیادی وجود دارند که در نظر گرفتن آن ها نامعقول خواهد بود. مثلا حالت هایی که حاشیه تمامی دستگاه ها موازی یکدیگر نباشد و یا حالاتی که روی صفحه ی نمایش همه دستگاه ها به یک سمت نباشد (مثلا یکی از دستگاه ها فیلم را برای جهت دیگر پخش نماید) پس می توان به جای پارامتر جهت گیری با سه متغیر پیوسته، از تعریف جدیدی به نام «چرخش» استفاده کرد.

چرخش: در پارامتر جهت اگر دو متغیر (از مجموع سه متغیر) را در بین تمامی دستگاه ها برابر با یک مقدار دلخواه و واحد در نظر بگیریم و متغیر سوم را تنها محدود به چهار حالت بنماییم (نماینده ی چهار حالت برای چرخش گوشی تلفن) آن را چرخش گوئیم.

چینش دلخواه

می توان به کاربران در انتخاب یک چینش اختیار داد، در این صورت مجموعه کاربران باید بر سر یک چینش به توافق برسند و سپس آن را به صورت فیزیکی با قرار دادن دستگاه ها کنار یکدیگر پیاده سازی نمایند. سامانه باید قادر باشد چینش پیاده شده در واقعیت را به تمامی پارامتر های آن (که در قسمت مدل سازی چینش تعریف شد) تبدیل نماید.

در این قسمت از سامانه باید برای به توافق رسیدن کاربران بر سر یک چینه راهکاری اندیشیده شود. این راهکار می‌تواند یک ساز و کار مستبدانه باشد (مثلاً حالتی که یک نقش یکتا باید صحت چینه را تایید کند) و یا می‌تواند از ساز و کار های تصمیم جمعی مانند رای‌گیری و اجماع استفاده نماید [5].

از مزایای چینه دلخواه این است که بار محاسباتی سامانه جهت دستیابی به یک چینه را می‌کاهد، چراکه کاربر این تصمیم را برای سامانه می‌گیرد.

چینه پیشنهادی

می‌توان قابلیت به سامانه افزود که با توجه به شرایط، چینه پیشنهادی برای دستگاه‌ها ارائه نماید. این ساز و کار در سامانه، چینه را با استفاده از پارامتر های زیر پیشنهاد می‌دهد:

- قاب فیلم
- صفحه نمایش حقیقی دستگاه‌ها

با مشخص بودن پارامتر های بالا می‌توان یک چینه پیشنهادی ارائه داد.

مراجع:

- [1] Hijazi, S., & Obaidat, M. S. (2019). Address resolution protocol spoofing attacks and security approaches: A survey. *Security and Privacy*, 2(1), e49.
- [2] The Robin Team, January 25 2018, *Moving to a New Office? Determine the Right Office Space Size*, robinpowered.com, <https://robinpowered.com/blog/new-office-determine-right-office-space-size#:~:text=The%20magical%20standard%20conference%20room%20size&text=Medium%20Conference%20Rooms10x25%20sq.,design%20with%20table%20and%20chairs>.
- [3] Wikipedia, *universally unique identifier*, wikipedia.org, https://en.wikipedia.org/wiki/Universally_unique_identifier
- [4] Electronic Systems Group, 2010, *MP3 decoder*, [es.ele.tue.nl](https://www.es.ele.tue.nl/dfbench/mp3decoder.php), <https://www.es.ele.tue.nl/dfbench/mp3decoder.php>
- [5] Garcia-Molina, H. (1982). Elections in a distributed computing system. *IEEE transactions on Computers*, 31(01), 48-59.
- [6] Mills, D. L. (2006). *Computer network time synchronization: the network time protocol*. CRC press.
- [7] Google Support Team, *share a mobile connection by hotspot or tethering on Android*, support.google.com, <https://support.google.com/android/answer/9059108?hl=en>
- [8] Han, B., Gopalakrishnan, V., Ji, L., & Lee, S. (2015). Network function virtualization: Challenges and opportunities for innovations. *IEEE communications magazine*, 53(2), 90-97.