 انجام این تمرین به‌صورت انفرادی است

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی امیرکبیر (پلی‌تکنیک تهران)

بسمه‌تعالی

دانشگاه صنعتی امیرکبیر (پلی‌تکنیک تهران)

دانشکده مهندسی کامپیوتر

درس مبانی اینترنت اشیاء

نیمسال دوم سال تحصیلی1400-1399

تمرین شبیه سازی سری چهارم

**مقدمه تمرین**

همان‌طور که می­دانید اینترنت اشیاء نقش به سزایی را در آسان سازی زندگی مردم، کاهش هزینه، صرفه‌جویی در مصرف انرژی و زمان ایفا می‌کند. آشنایی با پروتکل­های موجود و مورد استفاده در این تکنولوژی برای درک هرچه بیشتر آن الزامی است. حال هدف از انجام این تمرین، آشنایی با پروتکل­های MQTT و CoAP است، همان‌طور که می­دانید این دو پروتکل در کاربردهای اینترنت اشیاء بسیار محبوب هستند. از دلایل استفاده از این دو پروتکل به جای سایر پروتکل­های مرسوم مانند HTTP می­توان به کوچک­تر بودن هدر و بسته­های پیام­های ارسالی آن­ها اشاره کرد، بدیهی است که برای کاربردهای اینترنت اشیاء ما به بسته­های ارسالی کوچک­تری احتیاج داریم. این دو پروتکل جایگزین پروتکل مرسوم HTTP شدند. در دو لینک زیر می­توانید خلاصه­ای از مقایسه­ی این دو پروتکل با پروتکل HTTP را برای کاربردهای اینترنت اشیاء مشاهده کنید.

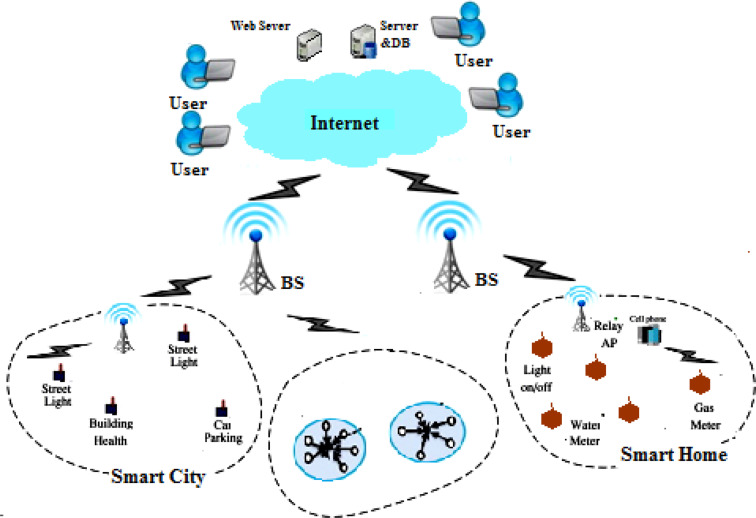
در این تمرین با کمک پروتکل­های ذکر شده به شبیه‌سازی کاربردهای اینترنت اشیاء در زندگی واقعی مانند شهر هوشمند، کشاورزی هوشمند و ... خواهیم پرداخت. لازم به ذکر است که انتخاب این کاربردها اختیاری است و شما می­توانید 3 کاربرد دلخواه را با ایده پردازی­های خود انتخاب و پیاده‌سازی کنید.

[مقایسه­ی پروتکل­های MQTT و HTTP برای کاربردهای اینترنت اشیاء](https://medium.com/mqtt-buddy/mqtt-vs-http-which-one-is-the-best-for-iot-c868169b3105)

[مقایسه­ی پروتکل­های CoAP و HTTP برای کاربردهای اینترنت اشیاء](https://www.rfwireless-world.com/Terminology/Difference-between-CoAP-and-HTTP.html)

**شرح تمرین**

همان‌طور که گفته شد اینترنت اشیاء تکنولوژی نوپایی است که در جهت هوشمند سازی هرچه بیشتر زمینه­های مختلف موجود در دنیا به کار می­رود. این زمینه­ها شامل شهر هوشمند، کشاورزی هوشمند، حمل‌ونقل هوشمند، موارد سلامتی و پزشکی هوشمند، پوشاک هوشمند و ... می‌شود. در این تمرین قصد داریم تا بر این کاربردها تمرکز کرده و با استفاده از پروتکل­های ذکر شده، این موارد را شبیه‌سازی نماییم. شما در پیاده‌سازی این تمرین باید سه کاربرد از کاربرد­های موجود را به‌دلخواه انتخاب کنید و سپس به پیاده‌سازی اجزای آن­ها بپردازید. مقصود از این امر پیاده‌سازی یک شبکه­ی heterogeneous برای کاربردهای اینترنت اشیاء است. برای اطلاع بیشتر از این مفهوم [اینجا](https://en.wikipedia.org/wiki/Heterogeneous_network) کلیک کنید. در شکل زیر مثالی از این نوع شبکه به تصویر کشیده شده است:



در ابتدا به توضیح کلی مراحل پیاده‌سازی می­پردازیم و پس از آن مراحل را به صورت خلاصه ذکر خواهیم کرد. در شکل زیر معماری مورد استفاده در این تمرین را مشاهده می­کنید:

Home1

Home10

…

Home2

P5

P1

…

همان‌طور که مشاهده می­کنید این معماری شامل یک subscriber، یک Broker و سه Publisher است. همان‌طور که گفته شد این سه Publisher را شما به صورت دلخواه انتخاب می­کنید. این Publisherها همان کاربردهای انتخابی شما هستند. برای مثال در اینجا خانه­های هوشمند، بیمارستان­های هوشمند و مزارع هوشمند انتخاب شده است. ابتدا باید بدانید که هرکدام از این Publisherها دارای حداقل 10 نمونه (instance) هستند. برای مثال در شکل بالا برای Home Node، ده خانه در ساختمان داده­ای دلخواه تعریف می­شوند. به همین ترتیب برای هرکدام از Farm Node و Hospital Node نیز ده نمونه تعریف می­گردد. هر کدام از این نمونه­ها حداقل دارای 5 پارامتر یا ویژگی (attribute) هستند. همان‌طور که در شکل نشان داده شده است همه­ی خانه­ها دارای 5 پارامتر (P1…P5) هستند که برای مثال در خانه­ها یک پارامتر به نام پارکینگ هوشمند داریم، وظیفه­ی این پارکینگ نشان دادن وجود یا عدم وجود ماشین در پارکینگ است. دیگر پارامتری که می­توان برای خانه نام برد سیستم گرمایشی آن است که دمای خانه را بازگو می‌کند. یک پارامتر دیگر در خانه می­تواند توان مصرفی برق خانه باشد. ( باید حداقل 5 پارامتر را در نظر بگیرید)

سپس شما برای این 10 خانه داده­هایی فیک می­سازید و آن را ذخیره می­کنید که برای ارسال به Subscriber توسط Broker مورد استفاده قرار می­گیرند. برای مثال در اینجا ما برای پارامترهای خانه­ها یک وضعیت فیک می­سازیم:

خانه 1 : پارکینگ : خالی ، دمای خانه: 27 درجه ، توان مصرفی : 50 وات

خانه 2 : پارکینگ : حالی ، دمای خانه: 23 درجه ، توان مصرفی : 60 وات

.

.

.

خانه 9 : پارکینگ : پر ، دمای خانه: 30 درجه ، توان مصرفی : 40 وات

خانه 10 : پارکینگ : خالی ، دمای خانه: 27 درجه ، توان مصرفی : 70 وات

*توجه داشته باشید همین روند برای بیمارستان و مزرعه و یا هر کاربرد دیگری که شما انتخاب می­کنید به کار گرفته می‌شود.*

حال به بررسی Subscriber می­پردازیم، در این بخش شما باید وب پیجی را جهت نمایش اطلاعات publisherها در Subscriber بالا بیاورید. این وب پیج در ابتدا سه دکمه با اسم کاربردهای انتخابی شما دارد که بر روی هر کدام از آن­ها که کلیک کنید اطلاعات آن بخش به نمایش گذاشته می‌شود. برای مثال اگر بر روی دکمه­ی Home کلیک شود 10 خانه­ی تعریف شده نمایش داده می‌شود و برای هر کدام از این 10 خانه به طور جداگانه 5 پارامتر نمایش داده می‌شود. یکی از این پارامترها را ( پارکینگ هوشمند) در بالا مثال زدیم.

لازم به ذکر است که در بخش MQTT، Publisherها باید ماشین­های مجازی مجزایی باشند و Subscriber کامپیوتر شخصی شما است. (Broker شما سرور آنلاین HiveMQ است)

در بخش CoAP، Broker نیز همانند Publisher­ها باید ماشین مجازی مجزایی باشد.

حال در ادامه به صورت خلاصه مراحل بازگو می­شوند:

1. ابتدا برای پیاده‌سازی این نوع شبکه شما باید حداقل سه کاربرد دلخواه خود را از زمینه­های موجود انتخاب کنید. ( برای مثال بیمارستان هوشمند و ...)
2. سپس در هرکدام از این کاربردها، حداقل 10 نمونه (instance) ذخیره کنید. برای مثال در کاربرد خانه­ی هوشمند شما باید 10 خانه را ذخیره کنید. سپس برای این نمونه­ها حداقل 5 پارامتر (attribute) را به‌دلخواه انتخاب کرده و شبیه‌سازی خود را بر پایه­ی آن­ها انجام دهید. برای مثال اگر خانه­ی هوشمند را انتخاب کردید یکی از پارامترهای آن می­تواند پارکینگ هوشمند باشد.
3. حال با توجه به معماری داده شده شما باید در نودهای کاربردهای انتخابی خود ( خانه­ی هوشمند و ...) برای هر پارامتر یا ویژگی، مقادیری تصادفی را بدست بیاورید و آن را در ساختمان داده­ای ذخیره کنید. برای مثال در خانه­ی هوشمند در بخش پارامتر پارکینگ هوشمند، ما ده خانه داریم که هر کدام یک پارکینگ هوشمند دارند و وضعیت وجود یا عدم وجود وسیله­ی نقلیه را ارسال می‌کند. این وضعیت­ها را به طور تصادفی مقدار دهی کنید.
4. حال اطلاعات هر کدام از نودهای Publisher را توسط Broker به Subscriber انتقال دهید.
5. با استفاده از یک وب پیج به گونه­ای که گفته شد اطلاعات را در Subscriber به نمایش بگذارید. توجه داشته باشید که در وب پیج شما ابتدا باید سه دکمه با اسامی کاربردهای انتخابی موجود باشد که با کلیک بر هرکدام از این دکمه­ها اطلاعات گرفته شده­ی تمامی پارامترهای آن زمینه ( مانند تمامی 10 خانه) به نمایش گذاشته می‌شود.

بخش اول : راه اندازی پروتکل MQTT

معماری ساختار MQTT شامل یک کارگزار (broker)است که با یک یا چند دستگاه ارتباط برقرار می‌کند. پایه ارتباطات بر اساس پیام­ها و تاپیک­هایی (subscribe/publish)است که به یک گره ارسال می‌شود. در این بین دیگر گره­ها نیز می‌توانند آن پیام را دریافت کنند. سرور مرکزی مسئول مدیریت شبکه و انتقال پیام­ها است.

1. برای استفاده از کارگزار MQTT شما از کارگزار آنلاین HiveMQ استفاده خواهید کرد. نحوه­ی کار با این کارگزار ساده است و با جستجو راجع به آن در اینترنت اطلاعات کافی را به دست خواهید آورد. نحوه­ی استفاده از این Broker را در سیستم خود شرح دهید.
2. ابتدا کاربردهای انتخابی خود را گزارش داده و پارامترهای انتخابی خود را توضیح دهید.
3. کتابخانه­های مورد نیاز برای client MQTT را نصب کنید و سناریو گفته شده را اجرا کنید. این پروتکل تقریبا برای اکثر زبان­های برنامه نویسی نسخه کلاینت دارد. پیشنهاد می‌شود از زبان python استفاده کنید. گزارشی راجع به نحوه­ی کار سناریوی خود تهیه کنید و کدهای خود را اجرا کرده و موردبررسی قرار دهید. همان‌طور که ذکر شد Publisherها باید ماشین­های مجازی مجزایی باشند و Subscriber نیز کامپیوتر شخصی شماست.
4. پارامترهایی که در بسته ارسالی از کلاینت به سرور و برعکس وجود دارد را توضیح دهید.

بخش دوم : راه اندازی پروتکل CoAP

همان‌طور که اطلاع دارید پروتکل CoAP بر مبنای درخواست/پاسخ (response/request)کار می‌کند و از متدهای Put, Post, Get و Delete برای دریافت اطلاعات استفاده می‌کند. در این بخش شما باید سناریوی تعریف شده را با استفاده از این پروتکل پیاده­سازی کنید.

1. بر خلاف بخش قبلی این بار کارگزار مورد استفاده­ باید توسط خود شما نوشته شود. این کارگزار را بر اساس پروتکل CoAP بنویسید. نحوه­ی کار Broker خود را شرح دهید.
2. کتابخانه­های مورد نیاز برای client CoAP را نصب کنید و سناریو گفته شده را اجرا کنید. این پروتکل تقریبا برای اکثر زبان­های برنامه نویسی نسخه کلاینت دارد. پیشنهاد می‌شود از زبان python استفاده کنید. گزارشی راجع به نحوه­ی کار سناریوی خود تهیه کنید و کدهای خود را ران کرده و موردبررسی قرار دهید. همان‌طور که ذکر شد Publisherها و Broker باید ماشین­های مجازی مجزایی باشند و Subscriber کامپیوتر شخصی شماست.
3. پروتکل CoAP و MQTT را از لحاظ معماری، مصرف انرژی، امنیت، کیفیت سرویس، سایز بسته ارسالی، با یکدیگر مقایسه کنید.

بخش سوم : راه اندازی پروتکل HTTP ( این بخش اختیاری است انجام این بخش نمره­ی امتیازی دارد)

پروتکل HTTP از دست تکانی (Handshaking)به منظور ارسال و دریافت اطلاعات استفاده می‌کند. در این روش برای شروع و پایان عملیات تبادل اطلاعات بین سرور و کلاینت، چندین درخواست و پاسخ رد و بدل می‌شود. لازم به ذکر است که در این فرآیند از پروتکل­های IP/TCP به منظور افزایش امنیت و تضمین ارسال داده­ها استفاده می‌شود.

1. بر روی سیستم شخصی server HTTP را نصب کنید و IP و Port مورد استفاده در server HTTP را نشان دهید. برای server HTTP پیشنهاد می‌شود از Nginx استفاده شود. (لازم به ذکر است که این سرور را خودتان نیز می­توانید پیاده‌سازی کنید.)
2. کتابخانه­های مورد نیاز برای client HTTP را نصب کنید و سناریو گفته شده را اجرا کنید. پیشنهاد می‌شود از زبان python استفاده کنید. گزارشی راجع به نحوه­ی کار سناریوی خود تهیه کنید و کدهای خود را اجرا کرده و موردبررسی قرار دهید. همان‌طور که ذکر شد Publisherها باید ماشین­های مجازی مجزایی باشند و Subscriber نیز کامپیوتر شخصی شماست.
3. دلایل جایگزین شدن پروتکل MQTT و CoAP به جای HTTP در بسیاری از کاربردهای اینترنت اشیاء را توضیح دهید.

بخش سوم : راه اندازی پروتکل AMQP ( این بخش اختیاری است انجام این بخش نمره­ی امتیازی دارد)

پروتکل AMQP یا Protocol Queuing Message Advanced یک استاندارد متن­باز برای انتقال پیام­های بین گره­ها است. معماری ساختار AMQP شامل یک brokerاست که با یک یا چند دستگاه ارتباط برقرار می‌کند. دو عملیات exchanges و queues برای انتقال اطلاعات از گره منتشرکننده‌ی پیام (Publish) به دریافت‌کننده‌ی پیام (Subscriber) در این پروتکل انجام می‌شود.

1. بر روی سیستم شخصی AMQP brokerرا نصب کنید. و IP و Port مورد استفاده در AMQP broker را نشان دهید. در این خصوص می­توانید از هر AMQP broker متن­باز استفاده کنید.
2. کتابخانه­های موردنیاز برای client AMQP را نصب کنید و سناریو گفته شده را اجرا کنید. گزارشی راجع به نحوه­ی کار سناریوی خود تهیه کنید و کدهای خود را اجرا کرده و موردبررسی قرار دهید. همان‌طور که ذکر شد Publisherها باید ماشین­های مجازی مجزایی باشند و Subscriber نیز کامپیوتر شخصی شماست.

**نحوه تحويل تمرین**

1. تحویل تمرین در قالب **7 فایل ویدئویی** انجام می‌شود، یعنی برای هر مرحله از 7 مرحله توضیح داده شده در بخش قبل باید یک فایل ویدئویی جداگانه وجود داشته باشد. در هر ویدئو مشخص شود کدام مرحله از مراحل فوق در حال انجام است. توجه داشته باشید که در هر ویدئو تمامی مراحل کار و نتایج به طور کامل **حداکثر در دو دقیقه** شرح داده شود.
2. در هر ویدئو باید مشخص شده باشد که این فایل متعلق به شما است. برای مثال قبل از توضیح مراحل انجام کار، **یک فایل word حاوی نام فرد، شماره دانشجویی و بخش مربوطه** بر روی سیستم نشان دهید که مشخص کند این ویدئو توسط شما ضبط شده است.
3. تمرین در **قالب یک فایل zip تحویل** داده شود و باید برای هر مرحله، ویدئو به همراه کد وجود داشته باشد. نحوه نام‌گذاری فایل باید به صورت زیر باشد:

**HW1\_StudentNumber.zip** که در آن StudentNumber شماره دانشجویی سرگروه است. (مثال: HW1\_9923110.zip)

1. دقت کنید که **حجم فایل Zip شده نهایی، حداکثر 150 مگابایت باشد**.
2. هر مرحله را به صورت زیر **نام‌گذاری** نمایید. این نحوه نام‌گذاری متناسب با تمرین خواسته شده در هر مرحله است.

* بخش اول (01-MQTT Broker, 02-MQTT usecase, 03-MQTT scenario, 04-MQTT packets)
* بخش دوم (01-CoAP broker, 02- CoAP scenario, 03- CoAP Comparison)

1. بخش‌های اختیاری این تمرین به‌صورت آنلاین تحویل گرفته می‌شود، زمان تحویل و قواعد آن متعاقباً اعلام

می‌شود.

1. تمامی ویدئو­های ضبط شده باید قابل پخش با آخرین نسخه نرم‌افزار KMPlayer باشد.
2. مهلت تحویل تمرین ۵ تیر ۱۳۹۹ است.
3. به‌ازای هر روز تأخیر ۵ درصد جریمه در نظر گرفته خواهد شد.
4. **در صورت عدم رعایت موارد ذکر شده، نمره مربوط به بخش خوانایی کسر خواهد شد.**

موفق و مؤید باشید