DOR :20.1001.1.2538161.1399.5.4.1.8

تاریخ دریافت مقاله: ۹۸/۰۶/۱۸
تاریخ پذیرش مقاله: ۹۹/۰۲/۰۶

تحلیل کاربرد الگوهای طراحی اینترنت اشیاء

زهرا بهمن پور
پست الکترونیکی zahra.bahmanpour1395@student.sharif.edu

رامان رامسین*
دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران
پست الکترونیکی: ramsin@sharif.edu

چکیده

الگوهـــا راه‌حل‌هایی هســـتند که افراد خبـــره برای حل مسائل تکرار شونده ارائه می‌دهند. در واقع الگو با ساختار ســـطح بالایی که از یک راه‌حل ارائه می‌دهـــد می‌تواند در حل مشـــکلات عینی مختلف مورد استفاده قرار گیرد. یکی از حوزه‌هایـــی که خصوصا در چند ســـال اخیر بســـیار مـــورد توجه قـــرار گرفته و محققان به اســـتخراج الگو در آن پرداخته‌اند، اینترنت اشـــیاء[1] اســـت. روز به روز اشیاء بیشـــتری در حال اتصال به شبکه جهانی هستند و در این میان سازمان‌ها و شرکت‌های مختلف هریک در حال ارائه راه‌حل‌های خود در این حوزه هســـتند؛ اما عدم یکپارچگی و وابســـتگی این راه‌حل‌ها به فنّاوری باعث شـــده تا قابل به‌کارگیری در کاربردهای مختلف نباشند. در این نوشتار سعی شده با مطالعه الگوهای طراحی مختلفی که تا به امروز در حوزه اینترنت اشیاء ارائه شده‌اند، مجموعه یکپارچه‌ای از ایـــن الگوها به همراه حوزه به‌کارگیـــری آن‌ها، در کنار هم ارائه شـــود. این الگوها طیف گســـترده‌ای از کاربردها مانند راه‌اندازی و ثبت، مصرف انرژی، ارتباطات، و امنیت دستگاه‌ها را پوشـــش می‌دهند. سپس این الگوها براساس مشابهت با هم، مکمل بودن برای هم، و نیازمندی‌های غیر کارکردی‌ای که برآورده می‌کنند، دسته‌بندی شده‌اند.

واژه‌های کلیدی: مهندســـی نرم‌افزار، الگوهای طراحی، اینترنت اشیاء، نیازمندی‌های غیرکارکردی.

۱- مقدمه

اینترنـــت اشـــیاء روز به روز در حال فراگیرتر شـــدن می‌باشد. از آنجایی که این مفهوم متعلق به صنعت یا حوزه خاصی نیســـت، راه‌حل‌های بســـیار زیادی در حوزه‌های مختلف مانند ســـلامت، حمل و نقل، و تولید ارائه شـــده‌اند، امـــا این راه‌حل‌ها پراکنده بوده و در برخی موارد به دنبال حل مســـائل مشابه و تکراری هســـتند؛ این مسئله، درک و استخراج مفاهیم بنیادین را دشوار می‌کند.

الگوها با ارائه راه‌حل‌های سطح بالا و عدم وابستگی به پیاده‌سازی یا فناوری خاص می‌توانند به احاطه بر مفاهیم بنیادین حوزه اینترنت اشیاء کمک کنند و ایجاد سیستم‌های اینترنت اشـــیاء را تســـهیل نمایند. هدف الگوهای طراحی، ارائه قطعات قابل اســـتفاده مجدد می‌باشد که از بُن‌سازه[2] و پیاده‌ســـازی مســـتقل باشـــند. اینترنت اشـــیاء از جمله

* نویسندهٔ مسئول

1- Internet of Things (IoT)

2- platform

حوزه‌هایی است که به دلیل توجه بسیار زیادی که در چند سال اخیر به خود جلب کرده است، انبوهی از فناوری‌ها و راه‌حل‌ها توسط مراکز پژوهشی و سازمان‌های مختلف برای آن ارائه شده‌اند. اما راه‌حل‌ها تا زمانی که به فناوری یا شرکت و سازمان خاصی وابستگی دارند شاید برای حل مشکل خاصی به‌کار آیند اما در درازمدت نمی‌توانند اثربخش باشند. شناسایی بیشتر الگوهای اینترنت اشیاء و دسته‌بندی آن‌ها به پژوهشگران و فعالان حوزه امکان می‌دهد تا در هنگام پیاده‌سازی یک سیستم اینترنت اشیاء بتوانند با سرعت و دقت بیشتری راه‌حل‌های خود را انتخاب و پیاده‌سازی نمایند.

در این مقاله، ابتدا مروری بر مولفه‌های اصلی تشکیل‌دهنده یک سیستم اینترنت اشیاء انجام شده و مهم‌ترین الگوهای بررسی شده در دسته‌های مجزا در جداولی آورده شده و شرح مختصری از هریک ارائه شده است. سپس به تحلیل و بررسی شباهت‌های میان الگوها و دسته‌بندی الگوهایی که بهتر است در مقام تکمیل یکدیگر مورد استفاده قرار گیرند پرداخته شده است. علاوه بر این با بررسی نیازمندی‌های غیرکارکردی که در حوزه اینترنت اشیاء مدنظر هستند به دسته‌بندی الگوها بر مبنای نیازمندی‌های غیرکارکردی‌ای که برآورده می‌کنند نیز پرداخته شده است.

از دسته‌بندی‌های ارائه شده در این پژوهش می‌توان برای طراحی و پیاده‌سازی یک سیستم یا برنامه کاربردی در حوزه اینترنت اشیاء استفاده کرد. ضمنا الگوهای هر حوزه می‌توانند نشانگر خوبی از وضعیت پیشرفت آن حوزه باشند و محققان می‌توانند بر اساس نتایج این پژوهش به سنجش و تصمیم‌گیری در مورد بخش‌هایی از حوزه اینترنت اشیاء بپردازند که نیاز به پژوهش بیشتر دارند.

بخش‌های بعدی این نوشتار به شرح زیر می‌باشند: در بخش ۲ به توضیح مفاهیم و کلیات اینترنت اشیاء پرداخته شده است؛ در بخش ۳ بررسی روی الگوهای اینترنت اشیاء انجام شده است و در بخش ٤ به تشریح یک زبان الگوی ارائه شده برای اینترنت اشیاء پرداخته شده است. بخش ٥ حاوی شرح نتایج این پژوهش و ارائه دسته‌بندی الگوهاست و بخش ٦ نیز حاوی جمع‌بندی و نتیجه‌گیری است.

۲- کلیات

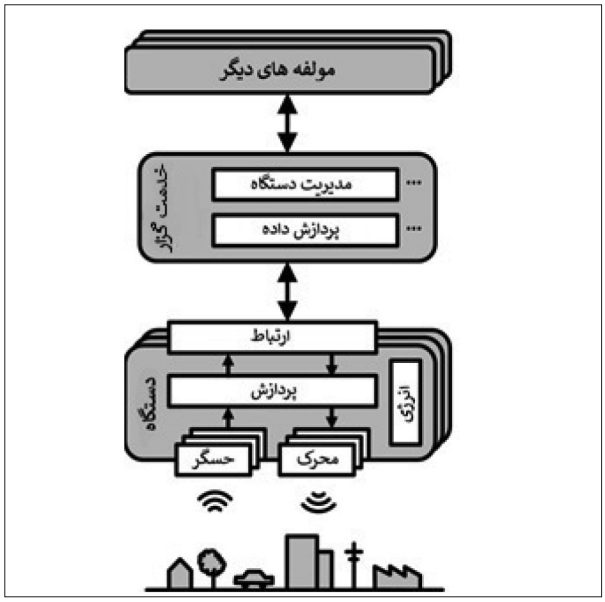
قبل از این‌که وارد بررسی الگوها شویم بهتر است نگاهی به مولفه‌های کلی یک سیستم اینترنت اشیاء داشته باشیم. معمولا اشیائی در مرکز یک سیستم اینترنت اشیاء قرار دارند که دارای حسگر، محرک، یک واحد پردازشی غالبا ضعیف، نوعی مسیر ارتباطی و یک منبع تامین انرژی هستند. حسگرها داده‌ها را از محیط دریافت کرده و آن‌ها را به سیگنال‌های الکتریکی تبدیل می‌کنند و محرک‌ها برعکس سیگنال‌های الکتریکی را به کنش‌های فیزیکی تبدیل کرده و از این طریق روی محیط اثر می‌گذارند. کل این مجموعه توسط یک واحد پردازشی کنترل می‌شود. هر دستگاه به‌صورت باسیم یا بی‌سیم با مولفه‌های دیگر، مثلا سایر دستگاه‌ها یا خدمت‌گزار یا فضای ابر، ارتباط برقرار می‌کند. معمولا یک خدمت‌گزار وجود دارد که جمع‌آوری داده و پردازش‌های سنگین‌تر را انجام می‌دهد. ساختار این مجموعه در شکل ۱ نشان داده شده است.

مولفه‌هایی که در شکل ۱ مشاهده می‌شوند در بسیاری از الگوها نام برده شده و برخی از آن‌ها مبنای ارائه الگوها قرار گرفته‌اند. برای مثال، منبع تامین انرژی دستگاه و یا چگونگی انجام پردازش‌های آن خود مجموعه الگوهایی دارد که در ادامه به آن‌ها پرداخته خواهد شد.

۳- بررسی الگوهای اینترنت اشیاء

در این بخش، الگوهای اینترنت اشیاء به همراه توضیحی مختصر، به تفکیک حوزه کاربرد آن‌ها، در قالب جدول‌هایی دسته‌بندی شده‌اند.

جدول ۱ در برگیرنده الگوهای دستگاه‌های اینترنت



شکل ۱: مولفه‌های سیستم‌های اینترنت اشیاء [۱]

اشـــیاء می‌باشـــد که هر یک مشـــکل خاصی را هدف قرار داده‌انـــد، و جـــدول ۲ شـــامل الگوهای راه‌انـــدازی و ثبت دستگاه‌های اینترنت اشـــیاء می‌باشد. دستگاه‌های موجود در فضای اینترنت اشـــیاء نیاز دارند تا از وجود یکدیگر و سرویس‌ها و راه‌های ارتباطیشان مطلع شوند. راه‌حل‌هایی برای این مســـئله ارائه شـــده اســـت، اما زیاد و گیج‌کننده بودن تعداد فنّاوری‌ها و اســـتانداردها انتخاب راه‌حل را به امری پیچیده تبدیل کرده اســـت [۲]. در [۲] سعی شده تا با بررسی راه‌حل‌های موجود، مجموعه الگوهایی جهت ثبت و راه‌اندازی دســـتگاه‌های اینترنت اشیاء ارائه شود. اصطلاح راه‌انـــدازی توصیف‌کننده فرایندی اســـت کـــه طی آن یک سیستم ساده، سیستم پیچیده‌تری را فعال می‌کند.

دسته بعدی الگوهای ارائه شده، الگوهای مدل دستگاه می‌باشد. در طول انجام عمل ثبت دستگاه، مدل دستگاه که نماینده مجازی آن اســـت ایجاد گردیده و در یک فهرست دســـتگاه ثبت می‌گردد. این فهرســـت شـــامل فراداده‌های بســـیاری راجع به دستگاه‌های ثبت شده می‌باشد که سایر دستگاه‌ها را قادر می‌سازد تا به جستجو و برقراری تعامل با این دستگاه‌ها بپردازند. این‌که این مدل دستگاه چگونه و کجا تعریف شود توسط دسته الگوهای مدل دستگاه مورد توجه قرار می‌گیرد [۲]. این الگوها در جدول ۳ ارائه شده‌اند.

جدول ۱: الگوهای دستگاه‌های اینترنت اشیاء

نام الگو	توضیح
Device Gateway	برخی دستگاه‌ها به دلیل عدم پشتیبانی فنّاوری ارتباطی، قادر به اتصال مستقیم به شبکه نیستند، و در نتیجه نیاز به اتصال از طریق یک درگاه مترجم[1] دارند [۱].
Device Shadow	دستگاهها و مولفه‌ها میتوانند همیشه با دستگاه تعامل داشته باشند؛ حتی وقتی دستگاه برخط نیست نیز یک نماینده مجازی از آن همیشه وجود دارد که هنگامی که دستگاه برخط شد با آن همگام می‌شود [۱].
Rules Engine	کاربران می‌توانند بدون نوشتن کد، قواعد ساده‌ای تعریف کنند که در هنگام وقوع یک رخداد اجرا شوند [۱].
Device Wakeup Trigger	به دستگاهی که به کارساز متصل نیست از طریق یک کانال ارتباطی کم مصرف اطلاع داده می‌شود تا اتصال را برقرار کند. دستگاه پیوسته در حال گوش دادن به آن کانال می‌باشد [۱].
Remote Lock and Wipe	زمانی که دستگاهی گم یا دزدیده شد، می‌توان از راه دور عملیات آن را غیرفعال و یا داده‌های روی آن را پاک کرد [۱].

1- Translating gateway

الگوهای منبع انرژی دسته دیگری هستند که در حوزه اینترنت اشیاء بسیار مورد توجه قرار می‌گیرند. این الگوها نشان‌دهنده منبع انرژی هستند که یک دستگاه ممکن است از آن اســـتفاده کند، که این امر به نوع دســـتگاه و شرایط محیطی آن هم بســـتگی دارد [۳]. ایـــن الگوها در جدول ٤ ارائه شده‌اند.

الگوهای دسته بعدی تعیین کننده حالت عملیاتی دستگاه هســـتند که عموما به میزان انرژی که دستگاه در هر لحظه در اختیار دارد بســـتگی دارنـــد [۳]. این الگوها در جدول ٥ آورده شده‌اند.

دســـته بعدی، الگوهایی هستند که روش‌های حسگری را مشخص می‌کنند. این الگوها تحت تاثیر نوع منبع انرژی دستگاه و حالت عملیاتی آن قرار می‌گیرند [۳]. الگوهای این دسته در جدول ٦ معرفی شده‌اند.

الگوهای مدیریـــت ارتباط الگوهایی هســـتند که نحوه برقراری اتصال بین دســـتگاه اینترنت اشیاء با خدمت‌گزار (یا سایر دســـتگاه‌ها) و تکنولوژی برقرار کننده ارتباط را شامل می‌شوند و در جدول ۷ آورده شده‌اند.

جدول ۲: الگوهای ثبت و راه‌اندازی دستگاه‌های اینترنت اشیاء

نام الگو	توضیح
Factory Bootstrap	اطلاعاتی که دستگاه برای برقراری اولین ارتباط با سایر دستگاه‌ها نیاز دارد در کارخانه در زمان ساخت درون آن قرار گرفته است [۲].
Medium-Based Bootstrap	اطلاعاتی که دستگاه برای برقراری اولین ارتباط با سایر دستگاه‌ها به آن‌ها نیاز دارد در زمان استقرار از طریق رسانه‌ای مانند یک دستگاه USB به آن انتقال می‌یابد [۲].
Remote Bootstrap	اطلاعاتی که دستگاه برای برقراری اولین ارتباط با سایر دستگاه‌ها به آن‌ها نیاز دارد از طریق یک کارساز راه دور به آن داده می‌شود [۲].
Automatic Client-Driven Registration	دستگاه برای این‌که خودش را به سایر دستگاه‌ها بشناساند، یک فرایند ثبت نام را آغاز می‌کند. در واقع یک API از کارساز را صدا زده و اطلاعات خود را به آن می‌دهد. این اطلاعات برای مراجعات بعدی، درون مدل دستگاه قرار می‌گیرد [۲].
Automatic Server-Driven Registration	کارساز از طریق سازوکاری از اضافه شدن دستگاه جدید باخبر می‌شود و با ارسال یک درخواست، اطلاعات دستگاه را از آن می‌خواهد. در نهایت پاسخ‌هایی که دریافت می‌کند را برای مراجعات بعدی درون مدل دستگاه نگهداری می‌کند [۲].
Manual User-Driven Registration	کاربر به‌صورت دستی عملیات ثبت دستگاه در کارساز را انجام می‌دهد. کاربر باید یک مدل دستگاه موجود را انتخاب کرده و یا از میان مدل‌های موجود انتخاب کند و سپس آن را با فراداده‌های مربوط به دستگاه پر نماید [۲].

جدول ۳: الگوهای مدل دستگاه

نام الگو	توضیح
Server-Driven Model	مدل‌های دستگاه در کارساز ایجاد و نگهداری می‌شوند. کارساز نمونه‌هایی از این مدل‌ها را به دستگاه‌های مربوطه اختصاص می‌دهد [۲].
Predefined Device-Driven Model	مدل‌های دستگاه در کارساز ایجاد و نگهداری می‌شوند. خود دستگاه‌ها مدلی را که با آن‌ها تطابق دارد را انتخاب کرده و آن را با فراداده‌های خود پر می‌کنند [۲].
Device-Driven Model	مدل دستگاه درون خود دستگاه ایجاد و نگهداری می‌شود و خود دستگاه این مدل را برای شرکای ارتباطی خود فراهم می‌کند[۲].

جدول ۴: الگوهای مصرف انرژی

نام الگو	توضیح
Mains-Powered Device	دستگاه به انرژی زیادی نیاز دارد و باید به یک منبع انرژی اصلی متصل شود. دستگاه می‌تواند پس از اتصال، ولتاژ بالا را به ولتاژ پایین تبدیل کرده و مصرف نماید [۳].
Period Energy-Limited Device	دستگاه به انرژی متوسطی نیاز دارد اما از نظر مکانی در جای دوری قرار دارد که به منابع انرژی اصلی دسترسی ندارد. یک منبع انرژی قابل تعویض یا قابل شارژ مجدد مورد استفاده قرار می‌گیرد و یک سازوکار هشداردهی در زمان پایین آمدن انرژی اخطار می‌دهد[۳].
Lifetime Energy-Limited Device	دستگاه به انرژی کمی نیاز دارد، سیار است یا در جای دوری قرار دارد. یک منبع انرژی در داخل دستگاه قرار میگیرد که در کل طول عمر دستگاه باقی خواهد ماند [۳].
Energy-Harvesting Device	دستگاه به انرژی کمی نیاز دارد، سیار است یا در محل دوری قرار دارد، و از نظر شرایط محیطی پایدار است. یک مولفه برداشت انرژی در داخل دستگاه قرار داده می‌شود تا انرژی موجود در محیط اطراف دستگاه مثل نور خورشید را به انرژی مصرفی آن تبدیل کند[۳].

دسته بعدی، الگوهای نوع پردازش در دستگاه‌های اینترنت اشیاء هستند که در جدول ۸ آورده شده‌اند.

طراحی برنامه‌های کاربردی اینترنت اشیاء تا حدی با سایر برنامه‌های کاربردی متفاوت است. دستگاه‌ها ممکن است با فنّاوری‌ها و شبکه‌های مختلفی به یکدیگر متصل باشند و محدودیت‌های پردازشی داشته باشند. در دسته بعدی الگوها سعی شده راه‌حل‌هایی در جهت آماده‌سازی محیط، مدیریت پیکربندی و استقرار و اندازه‌گیری سرویس

جدول ۵: الگوهای حالت عملیاتی

نام الگو	توضیح
Always-On Device	دستگاه منبع انرژی نامحدود دارد و باید همیشه در دسترس و پاسخگو باشد؛ در نتیجه همیشه روشن و متصل نگهداشته می‌شود [۳].
Normally-Sleeping Device	دستگاه منبع انرژی محدود دارد و باید انرژی مصرفی آن کمینه باشد، لذا طوری تنظیم می‌شود که مولفه‌های اصلی آن مادامی که به آن‌ها نیازی نیست غیرفعال باشند [۳].

جدول ۶: الگوهای روش حسگری

نام الگو	توضیح
Schedule-Based Sensing	حسگری دستگاه از قبل برنامه‌ریزی شده و دستگاه در بازه‌های زمانی مشخص عمل حس کردن محیط و ارسال اطلاعات به کارساز را انجام می‌دهد [۳].
Event-Based Sensing	حسگری دستگاه از محیط بر مبنای رخدادهایی است که در محیط اطراف اتفاق می‌افتند و یا از سمت کارساز برای آن ارسال می‌گردند [۳].

جدول ۷: الگوهای مدیریت ارتباط

نام الگو	توضیح
Delta Update	قصد داریم حجم پیام‌های حاوی داده حسگرها را بدون از دست دادن هیچ اطلاعاتی کم کنیم. برای این کار آخرین پیام ارسال شده را ذخیره کرده و دلتای آن را با داده فعلی حساب می‌کنیم. در نهایت فقط این دلتا و یک کد درهم‌ساز به دریافت‌کننده ارسال می‌کنیم و دریافت کننده پس از تطبیق داده مربوط به کد درهم‌ساز، دلتای دریافت شده را با داده خود ادغام می‌کند[۴].
Remote Device Management	هدف، مدیریت تعداد زیادی دستگاه از راه دور است. یک کارساز مدیریت راه‌اندازی کرده و دستگاه‌ها را تبدیل به مشتری میکنیم. کارساز مدیریت فرمان‌ها را به مشتری‌ها ارسال کرده و آن‌ها فرمان‌های مدیریتی را اجرا می‌کنند [۴].
Visible Light Communication	در یک محیط شلوغ نیاز به ارتباط بیسیم وجود دارد، اما نمی‌خواهیم از طیف رادیویی محیط استفاده کنیم. در نتیجه برای ارتباطات بیسیم نزدیک از طیف قابل رویت استفاده می‌کنیم و پیام‌ها را از طریق روشن و خاموش کردن سریع چراغ ارسال می‌نماییم [۴].

جدول ۸: الگوهای نوع پردازش

نام الگو	توضیح
Remote Processing	برخی از پردازش‌هایی که باید روی داده‌های جمع‌آوری شده دستگاه‌ها صورت بگیرد بسیار سنگین بوده و نیاز به قدرت پردازشی و ذخیره‌سازی بالایی دارند؛ در نتیجه باید در فضای دیگری مانند فضای ابر انجام گیرند و نتایج آن‌ها به دستگاه برگردانده شود [۵].
Local Processing	گاهی اوقات نیاز به پردازش سریع و واکنش فوری در مقابل برخی داده‌های خوانده شده توسط دستگاه‌ها وجود دارد؛ در نتیجه ارسال به فضای ابر و در انتظار پاسخ ماندن تاخیر زیادی به همراه خواهد داشت و بهتر است پردازش به‌صورت محلی روی خود دستگاه انجام پذیرد [۵].

جدول ۹: الگوهای طراحی برنامه‌های کاربردی اینترنت اشیاء

نام الگو	توضیح
Edge Provisioning	هدف، اطمینان از آماده بودن محیط و بستر اجرای برنامه در دستگاه‌ها می‌باشد؛ این کار می‌تواند از طریق مجازی‌سازی مبتنی بر مخزن[1] انجام شود [۶].
Edge Code Deployment	هدف، استقرار کد روی دستگاه‌های زیاد و به‌صورت سریع و کارا می‌باشد. کاربران کد خود را درون Git یا سایر سیستم‌های کنترل نسخه قرار داده و دستگاه‌ها مرتبا از یک نقطه مرکزی آخرین نسخه را درخواست می‌کنند [۶].
Edge Orchestration	هدف، هماهنگ کردن دستگاه‌های اینترنت اشیاء به‌عنوان گره‌های یک خوشه می‌باشد [۶].
Edge Diameter of Things (DoT)	هدف این است که ارائه‌دهندگان سرویس‌های اینترنت اشیاء میزان استفاده از واحدهای استقرار اینترنت اشیاء را محاسبه و ارزیابی کنند. برای این کار یک پروتکل اندازه‌گیری سبک وزن پیشنهاد شده است [۶].

1- Container-based Virtualization

جدول ۱۰: الگوهای پردازش مه

نام الگو	توضیح
Fogxy	شبیه الگوی Proxy از الگوهای GoF است، ولی پیشنهاد می‌کند که به جای هر ابر یک تکه ابر[1] به‌عنوان proxy قرار گیرد تا در موارد عدم دسترسی به ابر بتوان با Proxy آن ارتباط برقرار کرد [۷].
Fogging	هدف، ارائه سرویس‌های پردازشی، محیط ذخیره‌سازی و سایر خدمات به دستگاه‌های اینترنت اشیاء است. بُن‌سازه‌ای ارائه شده است که خدماتی مانند خدمات ابر ارائه می‌دهد اما تاخیر پایین، آگاهی از محیط و کارایی بالا را به همراه دارد [۸].

1- Cloudlet

جدول ۱۱: الگوهای امنیت در اینترنت اشیاء

نام الگو	توضیح
Secure Logger	رخدادهای کارساز درون فایل‌های وقایع نگار[1] به‌صورت رمزنگاری شده ثبت می‌شوند [۹].
Input Validation	ورودی کاربر سیار را اعتبارسنجی می‌کند[۹].
Secure Directory	حق دسترسی کاربر به فایل‌های درون کارساز را اعتبارسنجی می‌کند [۹].
Secure Adapter	به‌عنوان واسط بین خدمت‌گزار و بقیه دستگاه‌ها قرار می‌گیرد [۹].
Exception Manager	امنیت حالت‌های استثنا را تامین کرده و از افشای داده‌های حساس جلوگیری می‌کند[۹].

1- Log File

جدول ۱۲: الگوهای اینترنت اشیاء صنعتی

نام الگو	توضیح
Closed-Loop	استخراج و ذخیره اطلاعات فرایندها در فضای ابری برای تحلیل و بررسی بیشتر[۱۰].
Cloud-in-the-Loop	مانند Closed-Loop ولی بُن‌سازهٔ ابری به آن اضافه شده است[۱۰].
Open-Loop	فرمان‌ها به‌صورت یک طرفه از سمت سیستم کنترل به فرایندها فرستاده می‌شوند[۱۰].
Cloud-on-the-Loop	از نرم‌افزارهای مدیریت از راه دور روی ابر برای پیکربندی مجدد سیستم کنترل استفاده می‌شود [۱۰].
Publisher	تسهیل جمع‌آوری داده از دستگاه‌ها و انتشار داده‌های خوانده شده از محیط به بُن‌سازهٔ ابری [۱۰].
(Device-to-Device (D2D	بهره‌گیری از داده‌های مشترک بین دستگاه‌های همتا[۱۰].

برای برنامه‌ای کاربردی لبه ارائه شـود [٦]. این الگوها در جدول ۹ آورده شده‌اند.

پـردازش مه با هدف غلبه بـر محدودیت‌های پردازش ابری مانند تاخیر، بیدرنگی، آگاهی از محیط و ناهمگونی بُن‌سـازه‌ها ایجاد شـده اسـت. در جدول ۱۰ به بررسی الگوهایی که ایجادکنندگان برنامه‌های کاربردی پردازش مه از آن‌ها استفاده می‌کنند پرداخته شده است [۷]. پردازش مه در زمینه‌های بسیاری مانند اینترنت اشیا، واقعیت مجازی، و واقعیت افزوده کاربرد دارد.

از جمله مشـکلات حوزه اینترنت اشـیاء این است که بسـیاری از برنامه‌های کاربردی این حوزه بدون در نظر گرفتـن ملاحظات امنیتی خاص این سیسـتم‌ها طراحی و تولید شده‌اند [۹]. در جدول ۱۱، الگوهایی معرفی شده‌اند که اعمال آن‌ها می‌تواند امنیت را در سیسـتم‌های نرم افزاری اینترنت اشیاء تا حد مناسبی تامین کند.

وارد شـدن اینترنت اشـیاء در حوزه صنعت، انقلاب صنعتی بعدی خواهد بود که صنعت ٤/٠ نام گرفته اسـت. اینترنت اشیاء صنعتی[3] می‌تواند حوزه‌هایی مانند مدیریت کارخانه‌ها، بهینه‌سـازی فرایندها و خودکارسـازی تولید را پوشـش دهد. الگوهای این حوزه که در جدول ۱۲ ارائه شده‌اند تمرکزشان روی جریان داده در بین دستگاه‌ها در سیستم‌های کنترل صنعتی (ICS) می‌باشد [۱۰].

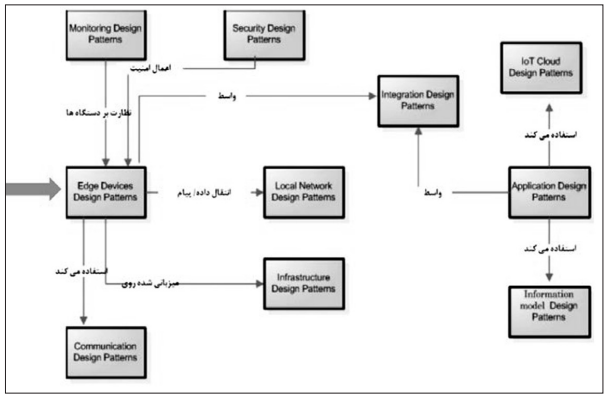
۴- زبان الگو برای اینترنت اشیاء

زبـان الگو روشـی برای توصیـف رویه‌های طراحی خوب در یک حوزه خاص می‌باشـد. عـلاوه بر این، زبان الگو می‌تواند نشـان دهنده وضعیت و میزان پیشرفت در

3- Industrial IoT

جدول ۱۳: الگوهای طراحی متعلق به زبان الگوی پیشنهادی در [۱۲] – متناظر با لایه‌های معماری اینترنت اشیاء

لایه معماری اینترنت اشیاء	الگوهای طراحی متناظر
مدیریت، یکپارچه‌سازی و کنترل دستگاه‌ها	Edge provisioning, Edge code deployment, Edge orchestration, Edge Diameter of Things (DoT)
ارتباطات	Request/Response, Event subscription, Asynchronous messaging, Reliable messaging, Multicasting, Publish/Subscribe, Queues, Message brokers, Federation, Discovery and delegation of trust- REST, Asynchronous events, Resource binding, Observer pattern, Publisher/Subscriber, Broker, Proxy, Protocol bridge, Resource discovery, Resource registration, Sleeping/Non-reachable endpoint
امنیت	Access control using data models, Social to physical graph relationship, PGP and asymmetric public-key cryptography on devices, DTLS over UDP, End-to-end encryption, Device management.
شبکه محلی	Peer-to-peer, Smartphone as gateway, Application gateway
یکپارچه سازی	Middleware platform
ابر اینترنت اشیاء	Cloud-based apps, Asset-based apps, Distributed IoT apps, Digital twin, Social IoT
زیرساخت	Mesh routing, Application gateway, Behind-NAT connectivity, M2M WAN
نظارت	Monitor a large number of devices over a large area
کاربرد	Virtualization, Virtualization by middleware, Software connected to thing via network, Thing-thing interaction
مدل اطلاعاتی	Structured data, Web Objects, Virtual objects-Smart objects, Composite objects, Hypermedia-HATEOAS, Semantic hyperlinks, Information model, Context model, Binding model, Resource directory - Catalog, Resource constructor, Access control model



شکل ۲: ترتیب اعمال الگوها بر روی برنامه‌های کاربردی مبتنی بر اینترنت اشیاء [۱۲]

یک حوزه تخصصی نیز باشد. فرایندی که توسط زبان الگو ارائه می‌شود رهنمودی دقیق و ملموس برای ایجاد سیستم در حوزه مربوط به زبان الگو ارائه می‌دهد. در [۱۲] دسته‌هایی از الگوهای طراحی، در کنار لایه‌ای از معماری اینترنت اشیاء که الگو در آن استفاده شده است، معرفی گردیده که آن‌ها را در جدول ۱۳ مشاهده می‌کنید. به الگوهای موجود در این جدول در قسمت‌های مختلف این نوشتار به‌صورت دقیق‌تر پرداخته شده است.

شکل ۲ نشان‌دهنده ترتیب‌هایی از الگوها است که بیانگر مسیرهایی برای اعمال الگوهای طراحی متعلق به زبان الگوی پیشنهاد شده در [۱۲] می‌باشند. مربع‌هایی که در این شکل مشاهده می‌شوند ، هر یک نماینده رده‌ای از الگوها (طبق دسته‌بندی جدول ۱۳) است و کمان‌های بین مربع‌ها بیان‌کننده ارتباطات بین این الگوها هستند. معمولا رده الگوهای طراحی دستگاه‌های لبه (Edge)، که در شکل با کمان بزرگ مشخص شده است، نقطه ورود به این مجموعه می‌باشد [۱۲].

جدول ۱۴: الگوهای مشابه

عنوان دسته	الگوها	شباهت
امنیت دسترسی	• Secure Directory • Access control using meta data	در هر دو الگو به نوعی دسترسی به داده‌ها یا فایل‌های یک دستگاه با بررسی مجموعه‌ای از فراداده‌ها صورت می‌گیرد.
مدیریت دستگاه	• Device Management • Device Registration and Device Model	هدف مشترک این الگوها استفاده از شناسه و لیست دستگاه‌ها در مدیریت آن‌ها می‌باشد.
اعمال قوانین	• Rule Oriented Programming • Rules Engine	الگوی اول برای نوشتن قوانین ماشین حالت برنامه‌ها مورد استفاده قرار می‌گیرد و الگوی دوم نیز به کاربر امکان نوشتن قوانین برای اجرای روی دستگاه‌های اینترنت اشیاء را می‌دهد.
پردازش از راه دور	• Remote Processing • Fogging	در الگوی اول به دلیل عدم وجود قدرت پردازشی لازم درون دستگاه نیاز به انجام پردازش در یک خدمت‌گزار راه دور وجود دارد. الگوی دوم نیز به دنبال انجام پردازش‌هایی است که قابل انجام روی خود دستگاه نیستند اما ارسال آن‌ها به ابر نیز از نظر تاخیر مقرون به صرفه نیست.
نماینده (Proxy)	• Fogxy • Device Shadow • Virtual Objects • Proxy	هر چهار الگو بر مبنای الگوی Proxy از مجموعه الگوهای GoF بنا نهاده شده‌اند و استفاده از یک نماینده مجازی را در زمانی که خود مولفه موردنظر قابل دسترسی نیست پیشنهاد می‌کنند.
مدیریت رخداد	• Event handler, onEvent • Event-Based Sensing	هر دو الگو بر مبنای رخدادهای ناهمگامی که رخ می‌دهند یک سری دستور را اجرا می‌کنند.
شناسایی منابع	• Resource Discovery • Resource Registration • Device Bootstrapping Patterns • Device Registration Patterns	مجموعه الگوهای راه‌اندازی و ثبت دستگاه که پیشتر به آن‌ها اشاره شد دقیقا به دنبال محقق کردن هدفی که الگوهای Resource Discovery و Resource Registration به دنبال آن هستند می‌باشند. این هدف، ثبت اطلاعات دستگاه در یک فهرست مرکزی و شناساندن دستگاه به سایر دستگاه‌های موجود است.
درگاه	• Device Gateway • Protocol Bridge	هر دو الگو به دنبال قرار دادن یک واسط برای تبدیل فنّاوری‌ها، پروتکل‌ها یا واسط‌هایی هستند که با هم منطبق نبوده و برای برقراری ارتباط با یکدیگر نیاز به تغییر دارند.
وضعیت دستگاه	• Normally-Sleeping Device • Sleeping/Non-reachable End-point	در هر دو الگو مولفه اصلی به‌صورت عادی در حالت خواب به سر می‌برد و فقط در صورتی که نیاز به برقراری ارتباط یا شروع یک تعامل داشت بخش‌هایی را که نیاز دارد فعال می‌کند.
ارتباط بین دستگاه‌ها	• Device-to-Device (D2D) • M2M WAN	الگوی اول از الگوهای اینترنت اشیاء صنعتی است که به پیاده‌سازی سیستم کنترلی با اتصال مستقیم دستگاه‌ها به یکدیگر اشاره دارد، و الگوی دوم نیز یک شبکه بی‌سیم ارتباط ماشین به ماشین است که عملا پیاده‌سازی الگوی اول می‌باشد.
شناسایی سرویس	• Edge Orchestration • Automatic Client-Driven Registration	در الگوی اول هدف هماهنگی و شناساندن سرویس‌هایی است که برنامه‌های اینترنت اشیاء می‌توانند ارائه دهند، و در الگوی دوم هدف شناسانده شدن سرویس‌های دستگاه‌هاست.

۵- دسته‌بندی الگوها

همان‌طور که دیده می‌شود، الگوهای اینترنت اشیاء متعددند و استفاده از آن‌ها بدون مدیریت پیچیدگی، مشکل است. دسته‌بندی الگوها می‌تواند استفاده از آن‌ها را برای طراحی و پیاده‌سازی یک سیستم یا برنامه کاربردی در حوزه اینترنت اشیاء تسهیل نماید. در این بخش، سه دسته‌بندی برای الگوهای طراحی اینترنت اشیاء ارائه شده است که نتایج اصلی این پژوهش می‌باشند.

۵-۱- دسته‌بندی الگوهای مشابه

در جدول ۱۴ نگاشتی بین برخی از الگوهایی که در بخش ۳ این نوشتار به آن‌ها پرداخته شد ارائه شده و

جدول ۱۵: الگوهای همراه (مکمل)

الگوها	علت
• Mains-Powered Device • Always-On Device • Device Gateway • Local Processing • Schedule-Based Sensing	از آنجا که دستگاه منبع انرژی بی‌پایان در اختیار دارد، نیازی به این‌که برای ذخیره کردن انرژی به حالت تعلیق درآید ندارد، و به دلیل در دسترس بودن همیشگی، برای تبدیل شدن به درگاه تبدیل فنّاوری‌های ارتباطی به یکدیگر ایده‌آل است. علاوه بر این، دستگاه با داشتن منبع انرژی می‌تواند پردازش‌های خود را به‌صورت محلی انجام داده و سربار تاخیر ارسال و دریافت از ابر را متحمل نشود و حسگرها نیز به‌صورت دوره‌ای داده‌ها را از محیط دریافت کنند.
• Energy-Harvesting Device • Normally Sleeping Device • Visible Light Communication • Device Wakeup Trigger • Remote Processing • Event-Based Sensing • Factory Bootstrap • Device Shadow	از آنجا که دستگاه انرژی موردنیاز خود را با برداشت انرژی از محیط مانند انرژی خورشیدی تامین می‌کند از این نظر بسیار محدودیت دارد؛ در نتیجه منطقی است که در اغلب موارد درحالت خواب قرار داشته باشد و فقط به یک کانال ارتباطی با مصرف انرژی کم گوش دهد تا در صورت نیاز قسمت‌های مختلف خود را فعال کند. از نظر روش برقراری ارتباط نیز کم مصرف‌ترین روش ارتباط، استفاده از نور مرئی می‌باشد. حسگری دستگاه نیز بهتر است بر مبنای وقوع رخداد باشد تا دستگاهی با انرژی محدود مجبور نباشد به‌صورت دائمی مرتبا به خواندن اطلاعات حسگر بپردازد. در نهایت اطلاعات راه‌اندازی نیز اگر به‌صورت پیش فرض و از کارخانه درون دستگاه وجود داشته باشد بهتر است، زیرا در اغلب موارد این‌گونه دستگاه‌ها در محل‌های دور حتی خارج از شهر استقرار می‌یابند و دسترسی به آن‌ها برای انجام راه‌اندازی اولیه می‌تواند مشکل باشد. از آنجا که دستگاه منبع انرژی و قدرت پردازش محدود دارد، امکان این‌که در زمان‌های مختلف در دسترس نباشد بسیار زیاد بوده و وجود یک دستگاه سایه (Device Shadow) که در موارد لزوم بتوان بجای خود دستگاه با آن ارتباط برقرار کرد مفید احساس می‌شود.
• Life-time Energy-Limited Device • Normally Sleeping Device • Remote Device Management • Remote Processing	دستگاهی که از نظر منبع انرژی در کل طول عمل خود با محدودیت مواجه است معمولا باید در حالت خواب قرار داشته باشد و فقط در صورت لزوم بخش‌هایی از آن فعال گردند. همچنین مدیریت دستگاه یا انجام عملیات پیکربندی و اعمال سیاست‌های مدیریتی و پردازش داده‌ها از راه دور برای این دستگاه مناسب‌تر است، زیرا در غیر اینصورت پیکربندی برعهده خود دستگاه و واحد پردازشی آن خواهد بود که از نظر مصرف انرژی به‌صرفه نیست.

چگونگی مرتبط بودن آن‌ها با هم نیز تشریح گردیده است. جهت سهولت اشاره به این الگوها در بخش‌های بعدی، عنوانی برای هر دسته در نظر گرفته شده است.

۵-۲- دسته‌بندی الگوهای همراه (مکمل)

الگوهای همراه، الگوهایی هستند که بهتر است در کنار یکدیگر به‌کار گرفته شوند و تا حد مطلوبی مکمل یکدیگر هستند. در جدول ۱۵ این الگوها به همراه تشریح چرایی مناسب بودن آن‌ها برای به‌کارگیری هم‌زمان گردآوری شده‌اند.

۵-۳- دسته‌بندی الگوها بر مبنای نیازمندی‌های غیرکارکردی

نیازمندی‌های سیستم‌های نرم‌افزاری عمدتا به دو دسته نیازمندی‌های کارکردی و غیرکارکردی تقسیم می‌شوند. نیازمندی‌های کارکردی آن دسته از نیازمندی‌ها هستند که مربوط به عملکردها، ویژگی‌ها و رفتاری هستند که یک سیستم باید از خود بروز دهد. نیازمندی‌های غیرکارکردی را می‌توان در واقع مجموعه محدودیت‌ها و کیفیت‌هایی دانست که باید روی نیازمندی‌های کارکردی اعمال شوند؛ مثلا سرعت پاسخ باید در حد معینی باشد و یا مصرف انرژی نباید از مقدار معینی تجاوز کند [۱۳]. یکی از چالش‌های اصلی طراحی سیستم‌های اینترنت اشیاء، برقراری نیازمندی‌های غیرکارکردی مانند طول عمر، مصرف انرژی، قابلیت اطمینان، توان عملیاتی، و تاخیر در آن‌هاست. از طرفی سیستم‌های اینترنت اشیاء معمولا با نیازمندی‌هایی روبرو هستند که باهم رقابت دارند و یا در برخی موارد ناقض یکدیگر هستند.

در این قسمت ابتدا به بررسی نیازمندی‌های غیرکارکردی که در یک سیستم اینترنت اشیاء مورد نظر هستند پرداخته شده و در ادامه الگوها بر مبنای نیازمندی‌های غیرکارکردی که برطرف می‌کنند دسته‌بندی شده‌اند. بدین ترتیب به ایجادکنندگان نرم‌افزارها و سخت‌افزارهای اینترنت اشیاء این امکان داده شده که بتوانند با توجه به نیازها و اولویت‌های خود از بین این

الگوهـــا انتخاب نمایند. نیازمندی‌هـــای عمده غیرکارکردی مرتبط با حوزه اینترنت اشیاء به قرار زیرند:

● **امنیت**: این نیازمندی به همراه حفظ حریم خصوصی، در زمره حیاتی‌ترین جنبه‌های اینترنت اشیاء محسوب می‌شود. در اینترنـــت اشـــیاء، برنامه‌های کاربـــردی از محیط، داده جمع‌آوری کرده و آن‌ها را جهت تحلیل به خدمت‌گزارهای مرکزی ارســـال می‌کنند که هم از نظر زیرپا گذاشته شدن حریم خصوصی و هم انـــواع حملات امنیتی مانند حملات مســـیریابی و حملات توزیع‌شده منع سرویس در معرض خطر قرار دارند [۱۳].

● **کارایـــی**: نیازمندی غیرکارکـــردی که تقریبـــا در کلیه سیســـتم‌های نرم‌افزاری باید تا حدی مدنظـــر قرار گیرد. مهم‌ترین مشخصه‌های کارایی، زمان پاسخ و توان عملیاتی می‌باشـــند که هرچـــه اولی پایین‌تر و دومی بالاتر باشـــد، کارایی مطلوب‌تر اســـت. برای رسیدن به محدوده مطلوب کارایی بایستی عوامل مختلفی را مدنظر قرار داد که به‌طور خـــاص در حوزه اینترنت اشـــیاء، کارایی شـــبکه، کارایی کارســـاز، کارایی دســـتگاه‌های متصل در شـــبکه اینترنت اشیاء و مواردی از این دست باید مدنظر قرار گیرند.

● **انعطاف‌پذیـــری**: انعطاف‌پذیری ویژگی سیســـتم‌هایی اســـت که دارای قابلیت تطابق با شـــرایط جدید در هنگام بروز تغییرات داخلی و خارجی هستند. این تغییرات اغلب از کنترل ایجادکنندگان و افراد خارج هستند. در حوزه اینترنت اشیاء که روز به روز انتظارات و توقعات کاربران در حال افزایش است، باید دســـتگاه‌ها، فناوری‌ها و استانداردهای ارائه شـــده قابلیت برآورده ســـاختن این نیازها را داشته باشـــند و بتوانند به‌صورت پویا خـــود را روزآمد کرده و کاربردهای جدیدی را به کاربران ارائه دهند.

● **بهینگی مصرف انرژی**: در کاربردهای اینترنت اشـــیاء غالبا با دســـتگاه‌هایی مواجه هستیم که از نظر منبع انرژی محدود بوده و بهینگی مصرف انرژی در آن‌ها مســـئله‌ای کلیدی می‌باشد.

● **قابلیت اطمینان**: منظور از قابلیت اطمینان این است که تا چه میزان سرویس‌هایی که توسط دستگاه‌ها در یک محیط اینترنت اشیاء ارائه می‌شود در دسترس بوده و می‌توان به درستی سرویس‌دهی آن‌ها اطمینان داشت.

● **ناهمگونی**: در یک فضای اینترنت اشـــیاء ممکن اســـت میلیون‌ها شـــیء وجود داشـــته باشـــند کـــه همگی قصد اتصال بـــه اینترنت و به یکدیگر را دارنـــد. در نتیجه انواع و اقســـام دســـتگاه‌ها از تلفن‌های همراه هوشمند تا لوازم خانگی و رایانه‌های شـــخصی در این فضا وجود دارند که هریک فناوری ارتباطی و قدرت پردازشـــی متفاوتی دارند و یکپارچه‌ســـازی این عدم تجانـــس باید قطعا مورد توجه ایجادکنندگان و فعالان حوزه اینترنت اشیاء قرار گیرد.

● **بیدرنگی**: کاربردهای اینترنت اشیاء در بسیاری موارد به گونه‌ای اســـت که میزان بیدرنگ بودن پاســـخی که به درخواستی داده می‌شود بسیار بااهمیت است، به‌طوری که در نتیجه عدم مطلوبیت بیدرنگی ممکن است جان انسان‌ها به خطر بیفتد؛ مثلا در مواردی که از حســـگرهای اینترنت اشـــیاء در دســـتگاه‌های تنظیم‌کننده ضربان قلب و سایر دستگاه‌های مربوط به سلامت انسان استفاده می‌شود در صورتی که بیدرنگی در ســـطح مطلوب نباشد عملا به‌کار نبردن دستگاه بر وجود آن برتری دارد.

● **قابلیت اســـتفاده**: هر فناوری و هـــر برنامه کاربردی چه در حوزه اینترنت اشـــیاء و چه خارج از آن باید دارای ســـهولت اســـتفاده بوده و روند به‌کارگیری ساده و قابل فهمی داشـــته باشد. در حوزه اینترنت اشیاء نیز این مسئله از اهمیت بالایی برخوردار است چرا که مخاطبان اینترنت اشیاء و برنامه‌های کاربردی آن ممکن است اعضای عادی جامعه، خردسال یا مسن باشند.

● **قابلیت مراقبت و نگهداری**: از آنجا که در برخی موارد ممکن است دستگاه‌های اینترنت اشیاء در محیط‌های خارج شهر و غیرقابل دســـترس قرار داشته باشند، بدیهی است که باید مراقبت و نگهداری از آن‌ها و برنامه‌های کاربردی روی آن‌ها به راحتی قابل انجام باشد، و به‌ویژه مدیریت از راه دور برای استفاده‌کنندگان میسر باشد.

جدول ۱۶: دسته‌بندی الگوهای اینترنت اشیاء بر مبنای نیازمندی‌های غیرکارکردی

نیازمندی غیرکارکردی	مجموعه الگوها
امنیت	Device Shadow, Remote Lock and Wipe, Secure Logger, Input Validation, Secure Directory, Secure Adapter, Exception Manager
کارایی	Device Shadow, Device Wakeup Trigger, Normally-Sleeping Device, Event-Based Sensing, Local Processing, Edge Code Deployment, Device to Device (D۲D)
انعطاف‌پذیری	Device Gateway, Device Shadow, Rules Engine, Medium-Based Bootstrap, Remote Bootstrap, Automatic Client-Driven Registration, Manual User-Driven Registration, Event-Based Sensing, Remote Device Management, Remote Processing, Edge Code Deployment, Publisher
بهینگی مصرف انرژی	Remote Bootstrap, Lifetime Energy-Limited Device, Energy-Harvesting Device, Normally-Sleeping Device, Schedule-Based Sensing, Event-Based Sensing, Remote Processing, Edge Diameter of Things (DOT)،
قابلیت اطمینان	Device Shadow, Device Wakeup Trigger, Remote Lock and Wipe, Factory Bootstrap, Pre-Defined Device-Driven Model, Mains-Powered-Device, Always-On Device, Edge Provisioning, Edge Orchestration, Fogxy, Secure Logger
ناهمگونی	Device Gateway, Medium-Based Bootstrap, Automatic Client-Driven Registration, Automatic Server-Driven Registration, Manual User-Driven Registration, Secure Adapter, Publisher
بلادرنگی	Device Wakeup Trigger, Automatic Client-Driven Registration, Mains-Powered-Device, Always-On Device, Event-Based Sensing, Local Processing, Edge Code Deployment
قابلیت استفاده	Rules Engine, Always-On-Device
قابلیت مراقبت و نگهداری	Device Shadow, Normally-Sleeping Device, Remote Device Management, Delta Update, Edge Provisioning, Edge code Deployment, Edge Orchestration, Edge Diameter of Things (DOT), Secure Logger
مقیاس‌پذیری	Device Gateway, Automatic Server-Driven Registration, Pre-Defined Device-Driven Model, Remote Device Management, Visible Light communication, Remote Processing, Edge Provisioning, Edge Code Deployment, Edge Orchestration, Fogging

● **مقیاس‌پذیری**: وجود میلیون‌ها شیء در جهان که تشکیل‌دهنده فضای اینترنت اشیاء هستند ثابت می‌کند که کلیه راه‌حل‌های ارائه شده در حوزه اینترنت اشیاء باید قابلیت پشتیبانی از تعداد بسیار بالای اشیاء (در حد نامحدود) و سر و کار داشتن با داده‌های حجیم تولید شده توسط این دستگاه‌ها را داشته باشند.

نیازمندی‌هایی که در قسمت بالا به آن‌ها اشاره شد در زمره نیازمندی‌های غیرکارکردی مهم و با اولویت بالا در حوزه اینترنت اشیاء هستند که علت اهمیت آن‌ها نیز تا حدی تشریح گردید. در جدول ۱۶ سعی شده تا الگوهای اینترنت اشیاء بر مبنای نیازمندی‌های غیرکارکردی دسته‌بندی گردند تا با توجه به هر نیازمندی بتوان مستقیماً به الگوهای مرتبط به آن مراجعه کرد.

کلیه الگوهایی که در این دسته‌بندی‌ها ارائه شدند می‌توانند با هر ترکیب دلخواهی با یکدیگر اعمال شوند، اما توجه به این نکته ضروری است که برخی از این الگوها با هم در تضاد هستند؛ مثلا دستگاهی که منطبق با الگوی Lifetime Energy-Limited باشد قطعا نمی‌تواند به‌طور مداوم و Always-On در حالت فعال قرار داشته باشد؛ یا دستگاهی که Device Gateway باشد منطقا نباید Normally-Sleeping باشد، زیرا سایر دستگاه‌ها برای برقراری ارتباط با یکدیگر به این دستگاه نیاز دارند. به‌طور کلی هر الگویی مزایا و معایب و محدودیت‌های خود را به همراه دارد و بسته به شرایط باید تصمیم گرفت که کدام الگو می‌تواند نتیجه بهتری برای کاربرد خاص ما داشته باشد.

۵-٤- ترتیب پیشنهادی برای استفاده از الگوها

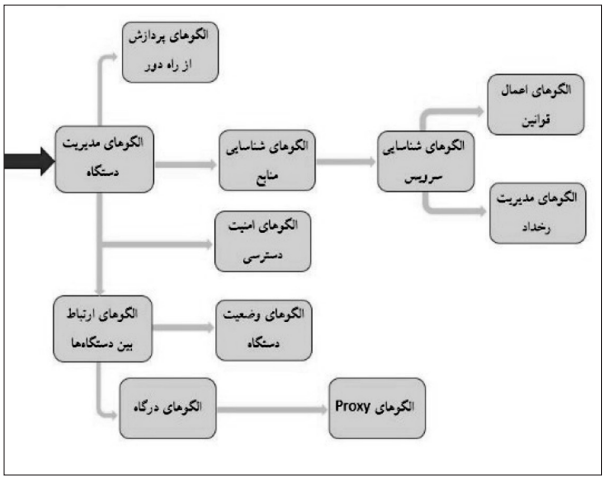
در این بخش ترتیبی برای اعمال دسته الگوهای جدول ۱٤ پیشنهاد شده است. ذکر این نکته ضروری است که این

ترتیب تنها با در نظر گرفتن سناریوهای عمومی معمول در محیط‌های اینترنت اشـیاء ارائه شده است؛ لذا در هر مورد خاص باید بررسی‌های بیشتری انجام گیرد تا تصمیم مقتضی در مورد ترتیب مناسـب اتخاذ گردد. علت این‌که در رابطه با اعمال الگوها برمبنای نیازمندیهای غیرکارکردی ترتیبی پیشنهاد نشـده این اسـت که اولویت اعمال نیازمندی‌های غیرعملکردی در سیسـتم‌ها و محیط‌های مختلف می‌تواند بسیار متفاوت باشـد؛ در نتیجه عملاً نمی‌توان برای اعمال الگوها بر اساس آن‌ها ترتیبی در نظر گرفت.

در شـکل ۳ ترتیب پیشـنهادی بـرای اعمـال الگوها مشاهده می‌شود. معمولا اولین دسـته الگویی که در نظر گرفته می‌شـود ناظر بر چگونگی مدیریت دسـتگاه‌ها در محیط اینترنت اشـیاء اسـت. در مدیریت دسـتگاه‌ها باید مشخص باشد که اعمال پردازشی آن‌ها قرار است در چه مکانی انجام گیرند. همچنین شناسـایی منابع و شناسایی سـرویس‌هایی که هر یک از منابع توان ارائه آن‌ها را دارد از دیگر موارد نیاز به الگوهاست. در شناسایی سرویس‌ها، چگونگـی اعمال قوانین بر دسـتگاه‌ها و مدیریت رخدادها مطـرح می‌گردنـد و لذا باید به سـراغ الگوهای مربوط به آن‌ها رفت. علاوه بر این، امنیت دسترسـی هم در مدیریت دسـتگاه‌ها و هم در ارتباط بین آن‌ها حائز اهمیت اسـت. در انتخـاب الگوهای ارتباط بین دسـتگاه‌ها باید در مورد وضعیت دسـتگاه اطلاع داشـت. علاوه بر این در رابطه با چگونگی ارتباط باید تصمیم گرفت و اگر تصمیم بر وجود واسـطی در برقراری ارتباط باشد، الگوهای درگاه مطرح خواهند شـد؛ ضمناً الگوهای نماینده (Proxy) میتوانند در انتخاب الگوهای درگاه در نظر گرفته شوند.

۶- نتیجه‌گیری

الگوهای طراحی راه‌حل‌هایی انتزاعی هسـتند که برای مسـائل تکرارشونده در حوزه‌های مختلف ارائه شده‌اند و در طول زمان ثابت شده که در حل مشکلات، مفید و موثر هستند. اینترنت اشـیاء مجموعه گسترده‌ای از دستگاه‌ها،



شکل ۳: ترتیب پیشنهادی برای اعمال دسته الگوهای مشابه

حسگرها، سرویس‌ها و به‌طور کلی اشیاء هوشمند متصل به هم می‌باشـد که داده‌هایشـان را به اشتراک می‌گذارند. در سـال‌های اخیر اینترنت اشیاء به حدی فراگیر شده که پیش‌بینی می‌شود تا سال ۲۰۲۰ حدود ۲۵ میلیارد دستگاه متصل در سراسر دنیا وجود خواهد داشت [۱۰].

از سـال‌های دور پژوهشـگران به اسـتخراج الگو در حوزه‌های مختلف مانند معماری نرم‌افزار و پردازش ابری پرداخته‌اند. از جمله حوزه‌هایی که بخصوص در سال‌های اخیر مـورد توجه قـرار گرفته و الگوهای آن اسـتخراج شده‌اند حوزه اینترنت اشیاء می‌باشد.

در ابتدای این نوشـتار، الگوهایی که در حوزه اینترنت اشـیاء تا به حال ارائه شـده‌اند بررسی شدند و سپس به دسـته‌بندی این الگوها پرداخته شد. حوزه‌هایی که الگوها برای آن‌ها ارائه شـده‌اند متفاوت بـوده و از منبع انرژی تا پردازش، ثبت و راه‌اندازی دسـتگاه، و امنیت را شـامل می‌شـود؛ اما نکته‌ای که قابل توجه اسـت این است که این الگوها هم از نظر تعداد نسـبت بـه حوزه‌های دیگر کمتر هسـتند و هم دارای کاسـتی‌هایی مانند عدم پوشایی و یا وابسـتگی به فنّاوری در برخی دسـته‌ها می‌باشـند، لذا همچنان نیاز به بررسـی و پژوهش بیشـتر در این حوزه احساس می‌شود.

در کارهای آتی قصد داریم به بررسـی بیشتر الگوها بپردازیم و دسته‌بندی‌هایی از ابعاد دیگر ارائه نماییم. علاوه

بر این، ارائه سناریوهای چگونگی استفاده از الگوها در کنار هم نیز از جمله کارهایی است که می‌تواند در ادامه این پژوهش انجام گیرد.

مراجع

1. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F., Riegg, A., “Internet of things patterns”, In Proceedings of the European Conference on Pattern Languages of Programs, pp. 5-26, 2016.
2. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F., Riegg, A., “Internet of Things Patterns for Device Bootstrapping and Registration”, In Proceedings of the European Conference on Pattern Languages of Programs, pp. 15-42, 2017.
3. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F., Riegg, A., “Internet of Things Patterns for Devices: Powering, Operating, and Sensing”, International Journal on Advances in Internet Technology, Vol. 10, No. 2, pp. 106-123, 2017.
4. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F., Riegg, A., “Internet of Things Patterns for Communication and Management”, In Transactions on Pattern Languages of Programming IV, pp. 139-182, 2019.
5. Reinfurt, L., Breitenbücher, U., Falkenthal, M., Leymann, F., “Applying IoT Patterns to Smart Factory Systems”, In Proceedings of the Advanced Summer School on Service Oriented Computing, pp. 1-10, 2017.
6. Qanbari, S., Pezeshki, S., Raisi, R., Mahdizadeh, S., Rahimzadeh, R., Behinaein, N., Mahmoudi, F., Ayoubzadeh, S., Fazlali, P., Roshani, K., Yaghini, A., “IoT design patterns: computational constructs to design, build and engineer edge applications”, In Proceedings of the International Conference on Internet-of-Things Design and Implementation (IoTDI), pp. 277-282, 2016.
7. Seitz, A., Thiele, F., Bruegge, B., “Focus Group: Patterns for Fog Computing”, In Proceedings of the European Conference on Pattern Languages of Programs, pp. 37-38, 2017.
8. Syed, M. H., Fernandez, E. B., Ilyas, M., “A pattern for fog computing”, In Proceedings of the Travelling Conference on Pattern Languages of Programs, pp. 13-22, 2016.
9. Lee, W. T., Law, P. J., “A case study in applying security design patterns for IoT software system”, In Proceedings of the International Conference on Applied System Innovation (ICASI), pp. 1162-1165, 2017.
10. Bloom, G., Alsulami, B., Nwafor, E., Bertolotti, I. C., “Design patterns for the industrial Internet of Things”, In Proceedings of the IEEE International Workshop on Factory Communication Systems (WFCS), pp. 1-10, 2018.
11. Umuhoza, E., Brambilla, M., “Model-driven Development of User Interfaces for IoT Systems via Domain-specific Components and Patterns”, In Proceedings of the International Conference on Enterprise Information Systems (ICEIS), pp. 246-253, 2017.
12. Chandra, G. S., “Pattern language for IoT applications”, In Proceedings of the Pattern Languages of Programs Conference, 2016.
13. Nguyen, X. T., Tran, H. T., Baraki, H., Geihs, K., “Optimization of non-functional properties in Internet of Things applications”, Journal of Network and Computer Applications, Vol. 89, No. 1, pp. 120-129, 2017.

جدیدترین کتاب
از انتشارات انجمن انفورماتیک ایران
منتشر شد!

تراوش‌های ذهنی

تهیهٔ کتاب از دفتر انجمن انفورماتیک ایران
(۶۶۴۱۲۸۶۱) و فروشگاه اینترنتی چاره
www.chare.ir

قیمت ۴۰/۰۰۰ تومان