



دانشکده مهندسی کامپیوتر و فناوری اطلاعات



آزمایشگاه اینترنت اشیا

گروه پلتفرم

عنوان سند:

# پیشنهاد طراحی، پیاده‌سازی، تست و تحویل پلتفرم اینترنت اشیا

کد سند:

IoT-PF-PM-0010-v1.00

تاریخ:

۹۶/۵/۸

کلیه حقوق هر نوع استفاده از این سند نزد آزمایشگاه اینترنت اشیا دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر محفوظ می‌باشد.

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

## اطلاعات سند

نام پروژه:	پلتفرم اینترنت اشیا
نام سند:	پیشنهاد طراحی، پیاده سازی، تست و تحویل پلتفرم اینترنت اشیا
کد سند:	IoT-PF-PM-0010-v1.00
وضعیت:	نهایی
تاریخ انتشار نهایی:	۱۳۹۶/۵/۸
نوع طبقه بندی سند:	محرمانه

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	ب

## فهرست مطالب

۱	فصل ۱: مقدمه
۴	فصل ۲: مشخصات و سوابق پیشنهاد دهنده
۵	۱-۲- معرفی پیشنهاد دهنده
۷	۲-۲- نمونه کارهای پیشین مشابه
۱۰	فصل ۳: راهکار پیشنهادی
۱۱	۱-۳- مقدمه
۱۲	۲-۳- پلتفرم اینترنت اشیاء
۱۳	۱-۲-۳- پلتفرم‌های موجود
۱۶	۳-۳- قابلیت‌های کارکردی پلتفرم پیشنهادی
۱۶	۱-۳-۳- قابلیت‌های ارتباط با اشیاء
۱۹	۲-۳-۳- قابلیت‌های ارتباط با برنامه‌های کاربردی
۲۱	۴-۳- قابلیت‌های غیر کارکردی سامانه پیشنهادی
۲۳	۵-۳- معماری سامانه پیشنهادی
۲۳	۱-۵-۳- معماری Microservices
۲۴	۲-۵-۳- معماری کلان پلتفرم پیشنهادی
۲۵	۶-۳- متدولوژی توسعه
۲۵	۱-۶-۳- متدولوژی Scrum
۲۷	۲-۶-۳- فرآیندهای انجام پروژه
۲۷	۲-۲-۶-۳- فرآیندهای سطح سامانه
۲۸	۳-۲-۶-۳- فرآیندهای سطح سرویس
۲۹	۷-۳- تخمین منابع سخت‌افزاری لازم برای میزبانی پلتفرم
۳۰	فصل ۴: ساختار سازمانی
۳۱	۱-۴- چارت سازمانی انجام پروژه
۳۲	۲-۴- وظایف واحدها
۳۵	فصل ۵: روش مدیریت و تضمین کیفیت
۳۶	۱-۵- کلیات طرح مدیریت پروژه
۳۶	۲-۵- کلیات طرح مدیریت پیکربندی
۳۶	۳-۵- کلیات طرح تضمین کیفیت
۳۷	۴-۵- کلیات طرح مدیریت ریسک

صفحه	تاریخ	کد سند
ج	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

۳۸ ..... ۵-۵ کلیات طرح مستندسازی

۳۹ فصل ۶: فازبندی و شرح خدمات

۴۰ ..... ۱-۶ فازبندی

۴۰ ..... ۲-۶ شرح خدمات و خروجی ها

۴۳ فصل ۷: زمان بندی

۴۵ فصل ۸: پوشش RFP

صفحه	تاریخ	کد سند
۵	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## فهرست اشکال

۱۴.....	شکل (۱-۳) معماری AWS IoT
۱۵.....	شکل (۲-۳) معماری Kaa
۲۳.....	شکل (۳-۳) معماری Microservices
۲۴.....	شکل (۴-۳) معماری کلان سامانه پلتفرم پیشنهادی
۲۶.....	شکل (۵-۳) متدولوژی Scrum
۲۷.....	شکل (۶-۳) فرآیندهای پروژه در سطح سیستم و سرویس
۳۱.....	شکل (۱-۴) چارت سازمانی انجام پروژه

صفحه	تاریخ	کد سند
۵	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## فهرست جداول

جدول (۱-۲) معرفی اعضای گروههای فعال در آزمایشگاه اینترنت اشیا.....	۹
جدول (۱-۳) تخمین سخت افزار مورد نیاز.....	۲۹
جدول (۱-۴) افراد کلیدی تیم پیشنهاد دهنده.....	۳۱
جدول (۱-۷) زمان بندی انجام پروژه.....	۴۴
جدول (۱-۸) نحوه پوشش نیازمندهای RFP توسط پیشنهاد پروژه.....	۴۶

صفحه	تاریخ	کد سند
و	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

# فصل ۱:

## مقدمه

صفحه	تاریخ	کد سند
۱	۹۶/۵/۸	IoT-PF-PM-0010-v1.00



ظهور اینترنت اشیا (IoT) به عنوان یک فناوری انقلابی که بسیاری از جنبه‌های زندگی بشری را تغییر خواهد داد، بستری برای راهکارهایی فراهم کرده است که در آن هوشمندی اشیاء نقش بسزایی دارند. از جمله این راهکارها می‌توان به شهر هوشمند، خانه‌های هوشمند، کشاورزی هوشمند، مدیریت انرژی هوشمند و غیره اشاره کرد. در اکو سیستم IoT، هوشمند بودن اشیاء شرط لازم است ولی تنها فاکتور تعیین کننده نیست. اشیاء هوشمند حجم بسیار بالایی از داده را از محیط حس می‌کنند. قدرت و ارزش اصلی راهکارهای مبتنی بر اینترنت اشیا در این نکته نهفته است که به چه نحوی بتوانند این داده‌ها را جمع‌آوری، تحلیل و بر اساس آن تصمیم‌سازی کنند. در معماری IoT، این مجموعه از وظایف بر عهده لایه پلتفرم است.

لایه پلتفرم به عنوان یک بستر برای مدیریت اشیاء، جمع‌آوری داده‌ها، تحلیل آنها و بستری برای توسعه برنامه‌های کاربردی است که بتوانند با دسترسی به این داده، تصمیمات لازم را اتخاذ کرده و به اشیاء ارسال کنند. در یک نگاه کلی، پلتفرم بی‌شباهت به سیستم عامل نیست چرا که همانند سیستم عامل می‌بایست جزئیات اشیاء را از برنامه‌های کاربردی پنهان کرده و امکان دسترسی برنامه‌های کاربردی برای اشیاء را فراهم کند. اما تفاوت مهمی هم بین سیستم عامل و پلتفرم وجود دارد. در IoT، پلتفرم نه با یک سخت‌افزار که با تعداد بسیار زیادی شی در تعامل است بنابراین مقیاس‌پذیری آن از اهمیت بالایی برخوردار است. اشیاء از طریق پروتکل‌های شبکه با پلتفرم در تماس هستند بنابراین امنیت این ارتباط مهم است. حسگرها داده کلان تولید می‌کنند که باید مدیریت شود. برنامه‌های کاربردی هم می‌توانند از راه دور به پلتفرم متصل شوند بنابراین API مناسبی برای این منظور لازم است.

علیرغم نقش کلیدی پلتفرم در راه کارهای مبتنی بر اینترنت اشیا، استاندارد مدونی در خصوص قابلیت‌ها و معماری این سامانه وجود ندارد. راهکارهای متعدد تجاری و متن باز در این خصوص ارایه شده است ولی با توجه به موارد ذیل نیاز به توسعه یک پلتفرم بومی و عام منظور وجود دارد.

- ♦ امکان‌ات سفارشی‌سازی پلتفرمهای متن باز محدود به معماری کلی آنها است.
- ♦ اتکاءپذیری پلتفرمهای موجود در کاربردهای حساس و حیاتی جای بحث و بررسی دارد.
- ♦ امنیت پلتفرمهای موجود برای همه کاربردهای حساس کفایت نمی‌کند.

برای تامین این هدف، پژوهشگاه فضایی ایران درخواست ارایه پیشنهاد برای توسعه این سامانه را دارد. پیشنهاد حاضر در پاسخ به این درخواست ارایه شده است. در این سند در ابتدا در فصل دوم، پیشنهاد دهنده

صفحه	تاریخ	کد سند
۲	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

معرفی شده و سوابق آن در حوزه اینترنت اشیاء شرح داده می شود. در فصل سوم، راهکار پیشنهادی برای طراحی و توسعه این سامانه ارایه شده است. در این فصل بعد از تبیین دقیق نیازمندی های کارکردی و غیرکارکردی سامانه در حوزه ارتباط با اشیاء و ارتباط با برنامه های کاربردی معماری این سامانه بر اساس الگوی معماری Microservices شرح داده می شود. سپس متدولوژی توسعه سامانه پیشنهادی که مبتنی بر Scrum است شرح داده شده و فرآیندهای مد نظر تشریح می گردد. در انتها نیز تخمین سخت افزار لازم برای میزبانی سامانه پیشنهادی ارایه می گردد. در فصل چهارم، ساختار سازمانی پروژه تشریح می گردد. فصل پنجم به تشریح روش مدیریت و تضمین کیفیت مورد استفاده در پروژه اختصاص دارد. فازبندی و شرح خدمات در فصل ششم آمده است و در نهایت زمانبندی پروژه نیز در فصل هفتم شرح داده شده است.

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۳

## فصل ۲:

# مشخصات و سوابق پیشنهاد دهنده

صفحه	تاریخ	کد سند
۴	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## ۲-۱- معرفی پیشنهاد دهنده

آزمایشگاه و گروه پژوهشی اینترنت اشیا به مدت دو سال است که با مشارکت ۸ نفر از اساتید دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر تأسیس شده است. هدف از تشکیل گروه مذکور حول اینترنت اشیا به عنوان یکی از محورهای اصلی پژوهشی دانشگاه، جهت دادن به فعالیت‌های آموزشی و پژوهشی بر اساس سیاست‌های کلان دانشگاه است تا با انجام هدفمند فعالیت‌های آموزشی، پژوهشی، توسعه فناوری و ایجاد شرکت‌های نوپای دانش‌بنیان علاوه بر حل مشکلات کشور، تولید دانش، ثروت و شغل نمود.

تیم پلتفرم که یکی از تیم‌های فعال گروه اینترنت اشیا دانشکده مهندسی کامپیوتر و فناوری اطلاعات است، توسط آقایان دکتر بخشی، دکتر صبایی و دکتر راستی مدیریت و راهبری می‌شود. تیم مذکور دارای تجربه مناسبی در طراحی، اجرا و نظارت پروژه‌های بزرگ نرم‌افزاری، شبکه و امنیت سامانه‌های اطلاعاتی و شبکه در کشور هستند که در رزومه قابل مشاهده است. تجربه تیم مذکور در هر چهار لایه اکوسیستم اینترنت اشیا شامل سخت‌افزار، شبکه، پلتفرم و برنامه‌های کاربردی باعث شده است که تمامی تخصص‌های لازم برای طراحی و پیاده‌سازی پلتفرم بومی در این تیم فراهم شده باشد.

در ادامه شرح مختصری از افراد کلیدی پروژه ارائه می‌گردد. رزومه افراد کلید پروژه، مشاوران و کارشناسان پروژه در ضمیمه ارائه می‌شود.

### دکتر مهدی راستی:

ایشان عضو هیات علمی دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر می‌باشد. زمینه‌های تخصصی ایشان عبارتند از تخصیص منابع در شبکه‌های بی سیم 5G، ارتباطات M2M و اینترنت اشیا و امنیت شبکه. مقاله منتشر شده ایشان در IEEE Communications Magazine در سال ۲۰۱۴ در زمینه 5G جز ده مقاله برتر IEEE در ماه‌های نوامبر و ژولای ۲۰۱۴ شناخته شده است (تعداد ارجاعات 300). هم چنین ایشان کارگاه‌های آموزشی در زمینه 5G در کنفرانس معتبر IEEE ICC 2016، در مرکز تحقیقات مخابرات ایران و دانشگاه صنعتی شیراز ارائه کرده‌اند.

از نظر صنعتی، ایشان مجری پروژه‌های مرتبط با امنیت شبکه در سطح ملی نظیر طراحی و پیاده‌سازی

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۵

سامانه شاخصهای امنیت فضای مجازی کشور، طرح کلان امنیت سیستم‌های کنترل صنعتی کشور و ارزیابی و تحلیل مخاطرات امنیتی سامانه‌های وب و پست الکترونیکی دستگاههای دولتی بوده و با موفقیت به اتمام رسیده‌اند. علاوه بر این، نظارت بر پروژه‌های مرتبط با امنیت شبکه در سطح ملی نظیر سامانه تولید ترافیک برای انجام تستهای عملکردی و پروژه ملی طراحی معماری مرکز عملیات امنیت را به عهده داشته‌اند. در آخرین پروژه صنعتی نیز ایشان بیش از دو سال است که از اعضای تیم مدیریت و راهبری پروژه ساختمان هوشمند است که به صورت عملیاتی در دانشکده مهندسی کامپیوتر در حال استفاده است.

### دکتر مسعود صبايي:

ایشان عضو هیات علمی دانشمده مهندسی کامپیوتر و فناوری اطلاعات می‌باشد. پروژه‌های شاخص مرتبطی که ایشان اجرا کرده‌اند به شرح ذیل است.

مدیریت پروژه مدیریت شبکه سوئیچ‌های بین‌شهری و سوئیچ‌های بین‌الملل: در این پروژه که بین سال‌های ۱۳۸۳ تا ۱۳۸۵ در شرکت پرورش داده‌ها انجام شده است، مدیریت ۱۱۰ مرکز سوئیچ بین‌شهری و بین‌الملل به صورت متمرکز در تهران در مرکز LCT مخابرات انجام شد. کلیه طراحی‌ها و پیاده‌سازی‌ها شامل مدل اطلاعاتی، تطبیق سوئیچ‌ها قدیمی، شبکه ارتباطی و برنامه‌های کاربردی مدیریتی در پنج حوزه مدیریتی توسط یک تیم فنی با بیش از ۶۰ نیروی متخصص در مدت دو سال انجام شد. حجم قرارداد این پروژه در سال ۱۳۸۳ پنج میلیارد و هفتصد میلیون تومان بود. که سهم نرم‌افزار آن یک میلیارد و نه صد میلیون تومان بود. این پروژه پس از گذشت ۱۰ سال هنوز مورد استفاده شرکت مخابرات است.

مدیریت پروژه سامانه‌های آموزش الکترونیکی (پروژه سامانه مدیریت یادگیری و کلاس مجازی): این پروژه در سال ۱۳۸۶ در مرکز آموزش الکترونیکی دانشگاه صنعتی امیرکبیر تعریف شد و در سال ۱۳۸۷ مورد بهره‌برداری قرار گرفت. این سامانه‌ها از سال ۱۳۸۷ تا کنون در دانشگاه صنعتی امیرکبیر و چندین دانشگاه و سازمان دیگر مورد استفاده قرار می‌گیرد. شرکت ایجاد شده برای ارائه این سامانه‌ها از معاونت علمی و فناوری ریاست جمهوری مجوز دانش‌بینان برای این محصول دریافت کرده است.

### دکتر بهادر بخشی:

دکتر بهادر بخشی، مقاطع تحصیلی کارشناسی را در دانشگاه صنعتی شریف و کارشناسی ارشد و دکتری

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۶

را در دانشگاه صنعتی امیرکبیر طی کرده و از سال ۱۳۹۱ نیز عضو هیئت علمی دانشکده مهندسی کامپیوتر دانشگاه صنعتی امیرکبیر می‌باشد. تخصص وی هم در زمینه پژوهش و هم صنعت، شبکه‌های کامپیوتری و نرم‌افزارها و محاسباتی مبتنی بر شبکه است.

دکتر بخشی تجارب متعدد صنعتی در حوزه توسعه نرم‌افزارهای شبکه و مشاوره و نظارت در این خصوص را دارد. طی سالهای ۱۳۸۵ الی ۱۳۹۰ ایشان مدیر تیم توسعه نرم‌افزار شرکت نیمه‌هادی سینا بودند که به توسعه نرم‌افزارهای نهفته در تجهیزات شبکه و نرم‌افزارهای مدیریت شبکه می‌پرداخت. طی سالهای ۱۳۸۸ الی ۱۳۹۲ ایشان مدیر پروژه و مدیریت تیم امنیت شبکه شرکت فاواگستر پارس بودند که عمده فعالیت این شرکت نظارت بر طراحی و پیاده‌سازی شبکه و نرم‌افزارهای مدیریت شبکه است. نظارت بر طراحی و پیاده‌سازی شبکه، امنیت و مدیریت شبکه IP/MPLS زیرساخت کشور از جمله پروژه‌هایی بودند که دکتر بخشی در آن مشارکت فعال داشت.

بعد از عضویت در هیئت علمی دانشگاه صنعتی امیرکبیر، علاوه بر اینکه حوزه پژوهش و تدریس دکتر بخشی متمرکز بر بحث‌های شبکه شده و چندین درس در حوزه توسعه نرم‌افزارهای مبتنی بر وب و مدیریت شبکه ارائه می‌کنند، ایشان مدیریت چندین پروژه صنعتی را نیز به عهده داشته‌اند. از جمله آنها می‌توان به مدیریت فنی پروژه توسعه فناوری‌های پایه IMS و مشاوره و نظارت بر راه‌اندازی سیستم تسویه حساب مالی اپراتورهای موبایل اشاره کرد که در این پروژه پردازش روزانه ۵ میلیارد CDR از اهداف اصلی است. در آخرین پروژه صنعتی نیز ایشان بیش از دو سال است که مدیریت تیم توسعه پلتفرم IoT را برعهده دارند که در طی آن پلتفرم اختصاصی هوشمندسازی ساختمان توسعه داده شد و به صورت عملیاتی در دانشکده مهندسی کامپیوتر در حال استفاده است.

## ۲-۲- نمونه کارهای پیشین مشابه

علاوه بر کارهای مرتبطی که هر یک از اعضای تیم پلتفرم در قالب پروژه‌های تشریح شده در رزومه انجام داده‌اند، این گروه در قالب یک تیم منسجم در آزمایشگاه اینترنت اشیا به مدت دو سال است که در حال اجرای پروژه‌های ذیل هستند.

♦ مدیریت هوشمند ساختمان: یکی از فعالیت‌های اصلی آزمایشگاه اینترنت اشیا طراحی و

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۷

پیاده سازی یک نمونه عملیاتی واقعی در ساختمان دانشکده مهندسی کامپیوتر است. در این پروژه، سیستم های هوشمند روشنایی و سرمایشی برای کل ساختمان دانشکده مهندسی کامپیوتر و همچنین نمایشگرهای هوشمند داخل کلاس های دانشکده بر اساس استانداردهای اینترنت اشیا طراحی و پیاده سازی شده است. همچنین در یک اتاق نمونه در داخل دانشکده علاوه بر سیستم های نمایشگر، روشنایی، و سرمایشی، سیستم های هوشمند اعلام و اطفاء آتش، هشدارهای گاز، گرمایشی، و حضور و غیاب به منظور مطالعه، پژوهش و توسعه در سطح کل دانشکده نمونه سازی شده است. این پروژه پایان یافته است و در صورت نیاز امکان بازدید از پایلوت پیاده سازی شده وجود دارد که در این پایلوت پلتفرم تهیه شده برای مدیریت هوشمند ساختمان قابل بررسی و ارزیابی توسط کارفرما وجود دارد.

♦ کنترل هوشمند پارکینگ: یکی دیگر از محورهای فعالیت آزمایشگاه اینترنت اشیا طراحی و پیاده سازی یک نمونه عملیاتی واقعی پارکینگ هوشمند در دانشگاه صنعتی امیرکبیر است. در فاز اول این برنامه، راه اندازی سیستم شناسایی و کنترل تردد وسایل نقلیه بر اساس سیاست های قابل تنظیم توسط مدیریت امور سازمانی و شهری هدف گذاری شده است. در این فاز تعدادی از نقاط شهری انتخاب شده به این سیستم مجهز خواهد شد. پایگاه اطلاعات مربوط به تردد وسایل قابلیت نگهداری حداقل تا شش ماه را دارد تا پردازش ها و تحلیل های هوشمند بر روی آنها برای استخراج قوانین رفتاری صورت گیرد. بدین ترتیب، امکان تحلیل رفتارهای سالانه، فصلی، هفتگی و ساعتی برای هر فرد یا وسیله در سیستم وجود دارد و سیاست گذاری متناسب با آن می تواند در سیستم تدوین گردد. سایت اطلاع رسانی طرح راه اندازی می گردد تا اطلاعات سیاست های جاری و نیز اطلاعات اختصاصی فرد/وسیله را برای استفاده مدیریتی و عام ارائه نماید.

♦ اندازه گیری هوشمند: هدف از این پروژه، طراحی و پیاده سازی یک بستر نمونه از سامانه ی هوشمند اندازه گیری و مدیریت مصرف کنتورهای آب، برق و گاز در ۲۰ ساختمان پردیس مرکزی و ۳ ساختمان پردیس اقماری (بندرعباس، ماهشهر، گرمسار) دانشگاه صنعتی امیرکبیر می باشد. در این پروژه علاوه بر هوشمند سازی کنتورهای آب، برق و گاز، برنامه کاربردی جهت نظارت و مدیریت مصرف انرژی در سمت مراکز ارائه دهنده انرژی و کاربران مصرف کننده طراحی و پیاده سازی می شود.

گروه های فعال در اینترنت اشیا، مسئول گروه ها و اعضای گروه معرفی شده است.

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۸

جدول (۱-۲) معرفی اعضای گروه‌های فعال در آزمایشگاه اینترنت اشیا

عنوان گروه	مسئول گروه
طراحی و پیاده‌سازی پلتفرم بومی اینترنت اشیا	دکتر بخشی
مدیریت هوشمند ساختمان	دکتر صبائی
اندازه‌گیری هوشمند	دکتر راستی
کنترل هوشمند پارکینگ	دکتر خرسندی

(در صورت نیاز، جزییات این محورها و شرح خدمات پروژه‌های فوق در مستند ۱۷۰ صفحه ای برنامه پیشنهادی آزمایشگاه اینترنت اشیا قابل ارایه است).

علاوه بر پروژه‌های فوق، موارد زیر نیز از فعالیتهای انجام شده گروه پلتفرم دانشکده مهندسی کامپیوتر و فناوری اطلاعات است که در کنار اجرای پروژه‌های فوق انجام گرفته است:

- ♦ از اسفندماه ۹۵، گروه پلتفرم این آزمایشگاه با تیم سخت افزار دانشکده مهندسی برق دانشگاه شریف به سرپرستی آقای دکتر عطاردی تفاهم‌نامه و موافقت‌نامه‌ای در خصوص همکاریهای متقابل منعقد کرده است. خروجی این همکاری، شروع پروژه پایلوت طراحی و پیاده‌سازی یک نمونه نیمه صنعتی سامانه و زیرساخت اندازه‌گیری هوشمند است.
- ♦ سلسله سیمنا‌های اینترنت اشیا بصورت ماهانه در دانشکده مهندسی کامپیوتر برگزار می‌گردد.
- ♦ انعقاد تفاهم‌نامه با صایران در سال ۹۵ در خصوص حمایت از پایان‌نامه مرتبط با اینترنت اشیا و امکان جذب ۱۰۰ نفر طی ۵ سال بصورت طرح کسر خدمت سربازی که پروژه‌های آنها مرتبط با اینترنت اشیا باشد (تاکنون ۱۵ پرونده کسر خدمت در خصوص انجام پروژه مرتبط با اینترنت اشیا بصورت فعال وجود دارد)



## فصل ۳:

# راہکار پیشنهادی

صفحه	تاریخ	کد سند
۱۰	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

### ۳-۱- مقدمه

در این فصل، راهکار پیشنهادی برای طراحی و پیاده‌سازی پلتفرم بومی عام منظوره تشریح می‌گردد که دارای چهار ویژگی اصلی است:

- ♦ این پلتفرم مختص یک راهکار و vertical خاص نیست بلکه در حوزه‌های مختلف مانند smart metering، کشاورزی هوشمند و شهر هوشمند و ... قابل استفاده است.
  - ♦ این پلتفرم وابسته به یک تکنولوژی خاص ارتباطی نیست بلکه برای ارتباط با اشیاء و برنامه‌های کاربردی از پروتکل‌های مختلفی پشتیبانی می‌کند.
  - ♦ این پلتفرم وابسته به یک نوع شی خاص نیست بلکه انواع اشیاء (حسگر و کنشگر) در قالب مدل‌های اطلاعاتی متنوع پشتیبانی می‌گردد.
  - ♦ این پلتفرم محدود به یک تعداد خاص از اشیاء نیست، بلکه دارای قابلیت مقیاس‌پذیری است.
- به منظور ارایه راهکار پیشنهادی، در ابتدا، قابلیت‌های فنی مد نظر در سامانه پیشنهادی شامل قابلیت‌های کارکردی و غیر کارکردی تدقیق می‌گردد. سپس فرآیندهای اصلی که بر اساس آنها، این سامانه طراحی، پیاده‌سازی، ارزیابی و مستندسازی می‌گردد، توضیح داده می‌شود. در ادامه این بخش از اصطلاحات و تعاریف ذیل استفاده خواهد شد:
- ♦ **شی هوشمند:** حسگر یا کنشگری که دارای عامل هوشمندسازی بوده و امکان تعامل با سایر اشیاء و پلتفرم را از طریق واسطه شبکه ارتباطی دارد.
  - ♦ **عامل هوشمند سازی:** نرم‌افزاری نهفته در اشیاء هوشمند که از طریق یک پروتکل ارتباطی با پلتفرم در تعامل است.
  - ♦ **مدل اطلاعاتی:** توصیف مستقل از پیاده‌سازی پارامترهای قابل مدیریت در یک شی است.
  - ♦ **زبان مدل سازی:** زبان که برای بیان مدل اطلاعاتی از آن استفاده می‌شود.
  - ♦ **SDK:** مجموعه ابزارهای توسعه نرم‌افزار شامل توابع و کتابخانه‌های کامپایل شده‌ای که تولیدکنندگان نرم‌افزار برای آسان کردن برنامه‌نویسی برای محیط یا سکوی خاصی فراهم می‌کنند و در اختیار برنامه‌نویسان کاربردی قرار می‌دهند.
  - ♦ **API:** رابط بین یک کتابخانه یا سیستم عامل و برنامه‌هایی است که از آن تقاضای سرویس می‌کنند.

صفحه	تاریخ	کد سند
۱۱	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

- ♦ **پروتکل لایه کاربرد:** بالاترین لایه در مدل TCP/IP که از سرویسهای لایه انتقال استفاده می کند.
- ♦ **ارتباط لایه کاربرد شی-پلتفرم:** تعاملات مابین اشیاء و پلتفرم که توسط پروتکل لایه کاربرد پشته پروتکلی پیاده سازی می شود.
- ♦ **ارتباط لایه کاربرد پلتفرم-کاربرد:** تعاملات مابین کاربردها و پلتفرم که توسط پروتکل لایه کاربرد پشته پروتکلی پیاده سازی می شود.
- ♦ **امکان get:** یکی از طرفین ارتباط بسته درخواست مبتنی بر خواندن یک پارامتر را ارسال کرده و طرف مقابل بسته پاسخ که مقدار پارامتر مد نظر است را ارسال می کند.
- ♦ **امکان set:** یکی از طرفین ارتباط بسته درخواست مبتنی بر تغییر یک پارامتر را به طرف مقابل ارسال کرده و طرف مقابل پس از تغییر، پاسخ مناسب را برمی گرداند.
- ♦ **امکان گزارش دوره ای:** اشیاء هوشمند بر اساس تنظیمات، به صورت دوره ای مقدار پارامترهای تعیین شده را به پلتفرم ارسال می کنند.
- ♦ **راهکار مبتنی بر اینترنت اشیاء:** راه حلی برای یک مشکل که از قابلیت های اینترنت اشیاء استفاده می کند، برای مثال کشاورزی هوشمند مبتنی بر اینترنت اشیاء.

### ۳-۲- پلتفرم اینترنت اشیاء

در معماری چهار لایه اینترنت اشیاء، پلتفرم به عنوان سکویی برای جمع آوری و تحلیل داده، مدیریت اشیاء و بستر توسعه برنامه های کاربردی از جایگاه ویژه ای برخوردار است. این سامانه از یک سو با اشیاء شبکه در ارتباط است که وظایفی از جمله دریافت اطلاعات مانیتور شده توسط حسگرها، ارسال دستورات به کنشگرها و مدیریت اشیاء انجام می دهد. از سوی دیگر پلتفرم در ارتباط با برنامه های کاربردی است که با مجردسازی جزئیات اشیاء و با فراهم آوردن سرویس های پایه امکان توسعه برنامه های کاربردی را ایجاد می نماید.

قابلیت هایی که در پلتفرم پیاده سازی شده است، تاثیر مستقیمی بر انعطاف پذیری، کارایی، مقیاس پذیری و قابلیت سفارشی سازی راهکار مبتنی بر اینترنت اشیاء (مانند شهر هوشمند، کشاورزی هوشمند و ...) دارد که از جمله آنها می توان به موارد ذیل اشاره کرد:

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۱۲

- ♦ پروتکل لایه کاربرد شی-پلتفرم پیاده‌سازی شده در پلتفرم که یک پروتکل اختصاصی است، نحوه تعامل اشیاء با پلتفرم را مشخص می‌کند. دستورات و کارکردهایی که در این پروتکل پیاده‌سازی می‌شود امکان قابلیت‌هایی مانند مانیتور کردن، ارسال دستورات و ... را فراهم می‌کند.
  - ♦ نحوه مجردسازی اشیاء که توسط یک مدل اطلاعاتی در پلتفرم انجام می‌شود تعیین‌کننده انعطاف‌پذیری و قابلیت سفارشی‌سازی و عمومی بودن راهکار را تعیین می‌کند. هر چقدر که این مدل اطلاعاتی و زبان توصیف آن جامع‌تر و عمومی‌تر باشد، امکان مدل‌سازی اشیاء مختلف و استفاده از آنها در راهکار مد نظر بیشتر خواهد بود.
  - ♦ طراحی معماری پلتفرم که عامل اصلی مقیاس‌پذیری این سامانه است قابلیت گسترش‌پذیری و توسعه راهکار مبتنی بر اینترنت اشیاء را تعیین می‌کند. اگر طراحی و پیاده‌سازی این سامانه دارای قابلیت مقیاس‌پذیری افقی نباشد با افزایش تعداد اشیاء در شبکه، این بخش تبدیل به گلوگاه شده و در نهایت به شکست راهکار مبتنی بر اینترنت اشیاء منجر خواهد شد.
  - ♦ پروتکل لایه کاربرد در پشته پروتکلی ارتباط پلتفرم-کاربرد که در حقیقت API پلتفرم برای برنامه‌های کاربردی است میزان سادگی و انعطاف‌پذیری برنامه‌های کاربردی را مشخص می‌کند. هر چقدر که این API تکمیل‌تر باشد توسعه برنامه‌های کاربردی و سناریوهای هوشمندسازی ساده‌تر و در نتیجه توانایی و قابلیت راهکار مبتنی بر اینترنت اشیاء بیشتر خواهد بود.
- مواردی از این دست نشان‌دهنده جایگاه کلیدی پلتفرم در راهکارهای مبتنی بر اینترنت اشیاء است. علیرغم اهمیت بالایی این سامانه، استاندارد و سند مدونی از قابلیت‌ها و معماری آن وجود ندارد. در حال حاضر راهکارهای مختلفی تحت عنوان پلتفرم اینترنت اشیاء ارایه می‌گردد که در ادامه مروری بر برخی از معروفترین آنها صورت گرفته است.

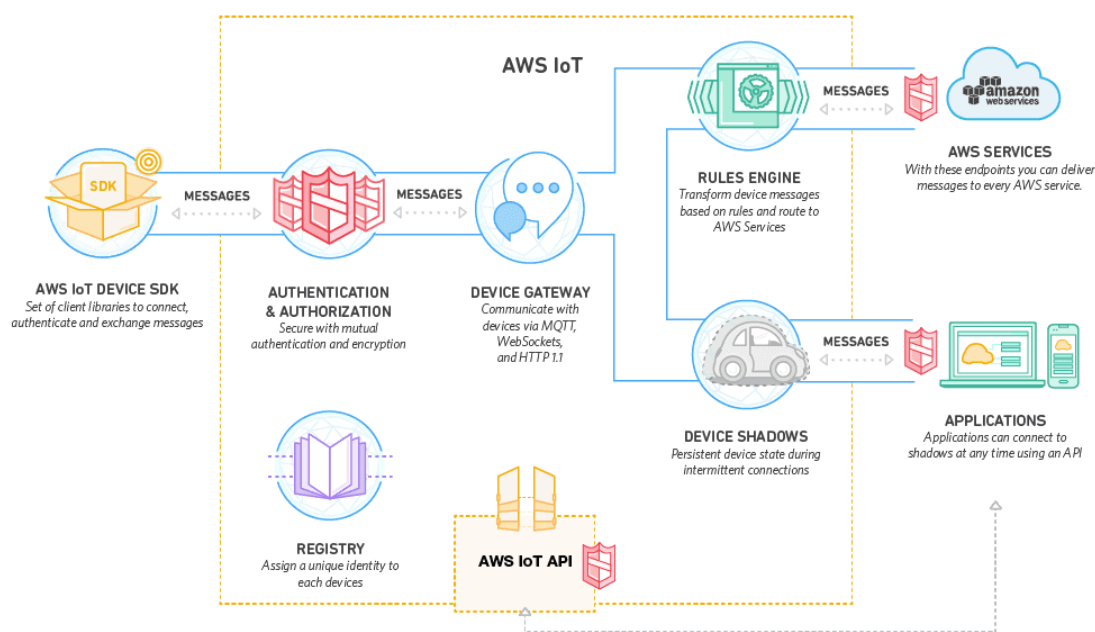
### ۳-۲-۱- پلتفرم‌های موجود

دو رویکرد کلی را می‌توان در پلتفرم‌های موجود مشاهده کرد. یک رویکرد با تمرکز بر توسعه برنامه‌های کاربردی، عموماً بر مدیریت اشیاء توسط یک برنامه کاربردی تمرکز دارند و برای این منظور امکانات متعددی از قبیل ثبت اشیاء، جمع‌آوری داده، API برنامه نویسی و ... را در اختیار قرار می‌دهد. در رویکرد دوم، بیشتر جنبه ارتباط ماشین با ماشین (M2M) مورد توجه قرار گرفته و پلتفرم ارتباط بین اشیاء را برقرار و مدیریت می‌کند.

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۱۳

از دیدگاه تجاری، عموماً دو نوع کلی راهکارهای پلتفرم‌های IoT وجود دارد. یک مجموعه از راهکارها، پلتفرم‌هایی هستند که خود پلتفرم در اختیار مشتریان قرار می‌گیرد تا در راهکار مد نظر نصب و راه‌اندازی شود. در مجموعه دوم راهکارها، خود پلتفرم در اختیار مشتری قرار نمی‌گیرد، بلکه امکانات پلتفرم در قالب سرویس‌هایی به مشتری ارائه می‌گردد. نرم‌افزارهای متن‌باز عموماً در دسته اول قرار دارند. حال آنکه راهکارهایی که توسط تامین‌کنندگان بزرگ ارائه می‌شود معمولاً در دسته دوم قرار دارند که در آن، پلتفرم اختصاصی در محیط ابر نصب شده و مشتریان می‌توانند اشیاء خود را به آن متصل کرده و برنامه‌های کاربردی مد نظر را در این بستر توسعه دهند.

پلتفرم AWS IoT محصول شرکت آمازون، یکی از محصولات معروف در دسته دوم است. این شرکت SDKهای لازم برای اتصال اشیاء را در اختیار تولیدکنندگان تجهیزات قرار می‌دهد و مشتریان می‌توانند با اتصال اشیاء خود به بستر ابر AWS از امکانات آن برای توسعه برنامه بر روی اشیاء خود استفاده کنند. معماری کلان این راهکار در شکل (۱-۳) نشان داده شده است.



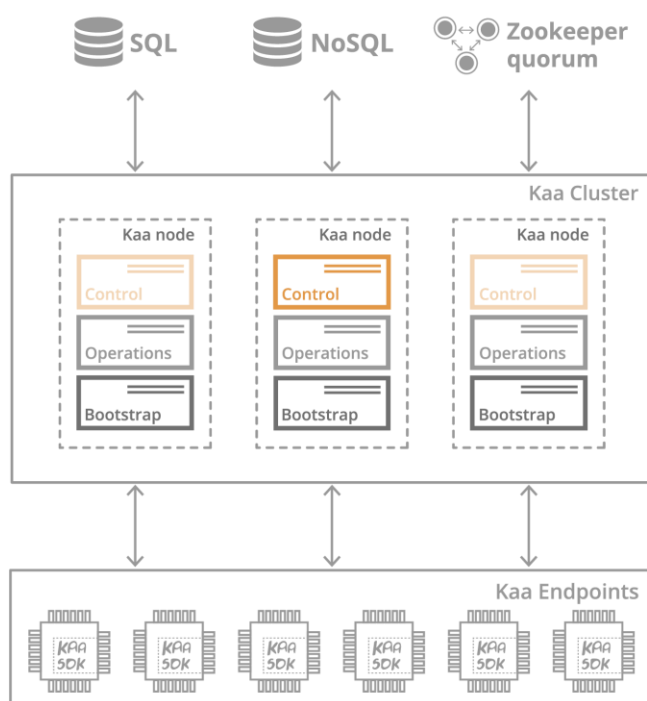
شکل (۱-۳) معماری AWS IoT

نمونه دیگری از پلتفرم‌های تجاری تحت بستر ابر Azure IoT شرکت مایکروسافت است. این راهکار نیز قابلیت‌هایی مشابه AWS IoT در اختیار قرار می‌دهد. پلتفرم اینترنت اشیاء شرکت ThingWorx یکی دیگر از این نمونه‌ها است. ادعای این پلتفرم در تسهیل توسعه برنامه‌های کاربردی است و علاوه بر امکانات پایه‌ای

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۱۴

پلتفرم، قابلیت تحلیل داده با مکانیزم‌های Big Data و روش‌های یادگیری ماشین را نیز در اختیار قرار می‌دهد. شرکت IBM هم محصول Watson IoT در این خصوص ارائه کرده است که امکانات کم و بیش مشابه AWS IoT را در اختیار توسعه دهندگان قرار می‌دهد. شرکت Cisco نیز ادعای ارائه سرویس پلتفرم را دارد و راهکار Cisco IoT Cloud Connect را ارائه کرده است. این محصولات، عموماً با رویکرد اول پلتفرم که بر توسعه برنامه‌های کاربردی برای کنترل و مدیریت اشیاء تمرکز دارد به مساله پرداخته‌اند. پلتفرم Carriots، محصولی است که رویکرد دوم یعنی تمرکز بر روی ارتباط M2M را مد نظر داشته و امکاناتی از قبیل مدیریت اشیاء، SDK توسعه و ... را برای این منظور در اختیار قرار می‌دهد.

در حوزه پلتفرم، محصولات و راهکارهای متعدد متن‌باز نیز وجود دارد که همانطوری که اشاره شد عموماً از جنس ابزارهایی است که مشتری می‌تواند در شبکه خود نصب و راه‌اندازی کرده و از آنها استفاده کند. یکی از این راهکارها، پلتفرم Kaa است البته لازم به ذکر است که نسخه جدید این محصول دیگر متن-باز نیست. این پلتفرم امکانات متعددی مانند SDK توسعه عامل مدیریتی، امکانات گزارش‌گیری و ... را در اختیار قرار می‌دهد که معماری آن در شکل (۲-۳) نشان داده شده است.



شکل (۲-۳) معماری Kaa

صفحه	تاریخ	کد سند
۱۵	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

پروژه‌های متعدد متن باز دیگری مانند OpenHAB، Kura، OpenIoT، و ThingSpeak نیز وجود دارد که ادعای ارایه پلتفرم برای اینترنت اشیاء را دارند. ولی حقیقت امر این است که همه این پروژه‌ها قابلیت‌های یکسانی و رویکرد مشابهی به بحث پلتفرم ندارد و همانطوری که ذکر شد به دلیل عدم وجود تعریف استاندارد از پلتفرم، هر یک از این ابزارها هدف خاص خود را دنبال کرده‌اند. با توجه به این نکته، قابلیت‌های کارکردی و غیر کارکردی که در این سامانه پیاده‌سازی می‌شود از اهمیت بسیار زیادی برخوردار بوده و باید به صورت دقیق شفاف‌سازی شود. علیرغم اینکه اکثر این موارد برای سامانه مد نظر، در RFP ذکر شده است ولی لازم است در ادامه این فصل به صورت دقیقتر شفاف‌سازی شده و مواردی که در RFP دیده نشده است به آن اضافه گردد.

### ۳-۳- قابلیت‌های کارکردی پلتفرم پیشنهادی

کارکردهایی که سامانه پیشنهادی در اختیار قرار خواهد داد را می‌توان به دو دسته کلی تقسیم‌بندی کرد. یک دسته از کارکردها در ارتباط با اشیاء شبکه است و دسته دوم مجموعه کارکردهایی است که برای توسعه برنامه‌های کاربردی بر روی بستر پلتفرم پیشنهادی، در این سامانه پیاده‌سازی خواهد شد. در ادامه این کارکردها شرح داده شده و برای هر یک، کدی اختصاص داده شده است که در جدول (۸-۱) مورد استفاده قرار می‌گیرد.

#### ۳-۳-۱- قابلیت‌های ارتباط با اشیاء

برای برقراری ارتباط با اشیاء یک پشته پروتکلی کامل در پلتفرم پیاده‌سازی می‌شود. البته واضح است که پیاده‌سازی لایه‌های فیزیکی (PHY) و کنترل دسترسی به رسانه (MAC) جزء وظایف لایه پلتفرم نیست. این پشته پروتکلی مجموعه قابلیت‌های ذیل را فراهم می‌آورد:

- ♦ در لایه PHY و MAC به یک پروتکل خاص وابسته نیست و امکان استفاده از پروتکل‌های NB-IoT و LoRaWAN برای این منظور وجود دارد. (FT01)
- ♦ علاوه بر پروتکل‌های استاندارد PHY و MAC در صورت نیاز به استفاده از پروتکل اختصاصی، پشته پروتکلی سامانه پیشنهادی امکان استفاده از آن را خواهد داشت. (FT02)
- ♦ پشته پروتکلی سامانه پیشنهادی برای ارتباط با اشیاء در لایه شبکه (Network) از پروتکل‌های

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۱۶

6LoWPAN و IPv6 پشتیبانی می کند. (FT03)

♦ پشته پروتکل سامانه پیشنهادی برای ارتباط با اشیاء در لایه کاربرد از پروتکل های CoAP و

MQTT پشتیبانی می کند. (FT04)

♦ طراحی پشته پروتکل سامانه پیشنهادی برای ارتباط با اشیاء به نحوی است که در صورت نیاز،

امکان استفاده از پروتکل های لایه کاربرد به غیر از دو مورد فوق الذکر نیز وجود دارد. (FT05)

♦ پلتفرم پیشنهادی، پروتکل مدیریت اشیاء اختصاص خود را دارد که بر روی پروتکل لایه کاربرد

ارتباط شی-پلتفرم پیاده سازی می شود. این پروتکل مدیریتی شامل دستورات زیر خواهد بود.

• امکان pull کردن اطلاعات توسط پلتفرم از اشیاء (قابلیت Get) که در آن

پارامترهای مد نظر با بسته درخواست ارسال شده و مقدار فعلی آنها در بسته پاسخ

برگردانده می شود. (FT06)

• امکان push کردن دوره ای اطلاعات از شی به پلتفرم (قابلیت گزارشهای دوره ای)

که در آن بدون اینکه پلتفرم درخواستی ارسال کرده باشد، حسگر هوشمند در

زمان های مشخصی اطلاعات مانیتور شده را برای پلتفرم ارسال می کند. این اطلاعات

به صورت مستمر در یک پایگاه داده ذخیره شده و در اختیار برنامه های کاربردی

قرار می گیرد. (FT07)

• امکان تغییر مقدار پارامترهای قابل مدیریت (قابلیت Set) که در آن پلتفرم

درخواست تغییر پارامترهایی را به همراه مقادیر جدید آنها ارسال می نماید و شی

هوشمند موفقیت آمیز بودن یا نبودن این تغییرات را در بسته پاسخ به پلتفرم اطلاع

می دهد. (FT08)

• امکان اعلام رخداد از شی به پلتفرم (قابلیت Trap/Notify) که در آن، در صورت

رخداد پیش آمدی خاصی، شی این رخداد را به اطلاع پلتفرم می رساند. این نوع

بسته ها عموماً نیاز به پردازش های سریع و بلادرنگ دارند و از این حیث با بسته های

push متفاوت هستند. (FT09)

♦ سامانه پیشنهادی امکان ثبت دستی و خودکار اشیاء را دارد. برای ثبت خودکار، یک بسته خاص

در پروتکل مدیریتی در نظر گرفته میشود که به صورت push عمل کرده و با boot شدن شی،

توسط عامل هوشمندی تعبیه شده در آن این بسته را به پلتفرم ارسال می نماید. اجزای این بسته به

نحوی است که اطلاعات لازم برای ثبت دستگاه و شناسایی آن توسط پلتفرم را در خود دارد.

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۱۷



برای ثبت دستی، نیاز به ارسال بسته مد نظر از طرف شی نیست، بلکه مدیر سامانه از طریق یک واسطه کاربر گرافیکی تحت وب، اطلاعات شی مد نظر را وارد کرده و آن شی در سامانه ثبت می گردد. در هر دو صورت در نهایت اطلاعات اشیاء ثبت شده در پایگاه داده مد نظر ذخیره می گردد. مجموعه اطلاعاتی که در این بخش برای شی در سامانه ثبت می شود (چه به روش خودکار و چه به روش دستی) توسط مدل اطلاعاتی این نوع از اشیاء مشخص می گردد. از این پس مدیر سامانه به صورت دستی و سرویس هایی که در پلتفرم پیاده سازی شده اند امکان دسترسی به این پایگاه داده و استخراج اطلاعات اشیاء را خواهند داشت. (FT10)

♦ مدیریت خرابی اشیاء: سامانه پیشنهادی امکان مدیریت خرابی اشیاء را برای مدیر سامانه فراهم می کند. برای این منظور، اطلاعات مربوط به رخ دادن خرابی که توسط عامل هوشمندسازی از طریق بسته های Trap به پلتفرم ارسال می گردد، در یک محیط گرافیکی به کاربر نشان داده می شود که مدیر سیستم با انتخاب هریک از آنها می تواند اطلاعات بیشتر در این خصوص بدست آورد. طبق بندهای RFP، جزییات این موارد در فاز طراحی نهایی سازی خواهد شد. (FT11)

♦ مدیریت پیکربندی اشیاء: سامانه پیشنهادی برای مدیریت پیکربندی اشیاء اولاً امکان خواندن (از طریق بسته های pull) و تغییر (از طریق بسته های set) پارامترهای مدیریت اشیاء را فراهم می نماید. ثانیاً با نگهداری (حداقل یک) نسخه پشتیبان از پیکربندی اشیاء، این امکان را فراهم می کند تا در صورت تعویض یک شی با شی جدید، پیکربندی آن به سهولت انجام گیرد. طبق بندهای RFP، جزییات این موارد در فاز طراحی نهایی سازی خواهد شد. (FT12)

♦ مدیریت حسابرسی اشیاء: سامانه پیشنهاد برای مدیریت حسابرسی اشیاء، این امکان را فراهم می کند که اطلاعات مربوط به استفاده از این شی در پایگاه داده ثبت گردد. این اطلاعات میتوانند توسط برنامه های کاربردی برای انجام محاسبات حسابرسی مورد استفاده قرار گیرند. طبق بندهای RFP، جزییات این موارد در فاز طراحی نهایی سازی خواهد شد. (FT13)

♦ مدیریت کارایی اشیاء: سامانه پیشنهادی امکان مدیریت کارایی اشیاء را در یک محیط گرافیکی برای مدیر سامانه فراهم می نماید. برای این منظور اطلاعات کارایی اشیاء به صورت متناوب جمع آوری در پایگاه داده های مناسبی ذخیره می شود. این اطلاعات به صورت گرافیکی در قالب نمودارهای مختلف می تواند به کاربر نشان داده شود. از سوی دیگر با مقایسه این اطلاعات با حد آستانه های تعریف شده، امکان ایجاد اعلان هشدار برای عبور از این حد آستانه ها وجود دارد. طبق بندهای RFP، جزییات این موارد در فاز طراحی نهایی سازی خواهد شد. (FT14)

♦ مدیریت امنیت اشیاء: سامانه پیشنهادی امکان مشاهده و تغییر پارامترهای مربوط به امنیت شی مانند

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۱۸

(کلمات عبور و پروتکل های امنیتی) را برای مدیر سامانه فراهم می نماید. طبق بندهای RFP، جزییات این موارد در فاز طراحی نهایی سازی خواهد شد. (FT15)

♦ داده هایی که توسط حسگرها تولید شده و به پلتفرم ارسال می گردد (عموما از طریق مکانیزم push دوره ای و یا در برخی از موارد به صورت pull) توسط سامانه پیشنهادی در پایگاه داده مناسب ذخیره می گردد. (FT16)

♦ پیاده سازی قابلیت های فوق الذکر در پلتفرم به تنهایی کافی نیست بلکه لازم است تا قابلیت های متناظر در اشیاء و درگاه های شبکه نیز پیاده سازی شود. برای این منظور سامانه پیشنهادی امکان تولید کتابخانه و SDK برای عامل های هوشمندسازی را در اختیار قرار می دهد. این کتابخانه می تواند برای سیستم های عامل متنوعی کامپایل شده و در کنار عامل هوشمندسازی مورد استفاده قرار گیرد. مهمترین وظیفه این کتابخانه ارایه یک API به عامل هوشمندسازی است تا بدون نیاز به پیاده سازی پشته پروتکلی و با استفاده از این کتابخانه به پلتفرم متصل شود. (FT17)

### ۳-۲- قابلیت های ارتباط با برنامه های کاربردی

وظیفه اصلی پلتفرم در قبال برنامه های کاربردی، ایجاد بستری است که بتوان این برنامه ها را به سادگی توسعه داد. این وظیفه از طریق فراهم آوردن API های مناسب برای دسترسی به اشیاء و داده های آنها انجام می گیرد. مشخصا برای این منظور سامانه پیشنهادی قابلیت های زیر را در اختیار قرار می دهد.

♦ پشته پروتکل ارتباطی مابین پلتفرم و برنامه های کاربردی در لایه ۳ و ۴ و ۵ از پروتکل های استاندارد مانند IPv4، IPv6، TCP، UDP و HTTP پشتیبانی می کند. (FT01)

♦ دسترسی به اشیاء و داده های آن در قالب API برای برنامه های کاربردی از طریق یک پروتکل اختصاصی که بر روی پروتکل لایه کاربرد ارتباط پلتفرم-کاربرد پیاده سازی می شود، صورت می پذیرد. این پروتکل امکانات ذیل را برنامه های کاربردی فراهم می نماید. (FA02)

- برنامه های کاربردی از طریق این API امکان خواندن مقدار پارامترهای مدیریت اشیاء را در اختیار خواهند داشت. (FA03)

- برنامه های کاربردی می توانند از طریق این API مقدار پارامترهای قابل تغییر مدیریت اشیاء را تغییر دهند. (FA04)

- از طریق این پروتکل مدیریت، پلتفرم می تواند رخدادهای پیش آمده را به برنامه های کاربردی گزارش نمایند تا مورد پردازش قرار گیرند. (FA05)

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۱۹

♦ با توجه به اینکه ارتباط برنامه‌های کاربردی با پلتفرم به صورت یک سرویس در سامانه پیاده‌سازی می‌گردد، سامانه پیشنهادی قابلیت اضافه نمودن هر پروتکل لایه کاربرد اختصاصی که سازگار با TCP/IP باشد را دارد. (FA06)

♦ به منظور توسعه برنامه‌های کاربردی بدون وابستگی به جزییات پیاده‌سازی اشیاء، سامانه پیشنهادی اشیاء را در قالب مدل‌های اطلاعاتی مجردسازی و در اختیار برنامه‌های کاربردی قرار می‌دهد. این مدل‌های اطلاعاتی از طریق یک واسطه کاربر گرافیکی قابل ایجاد، مشاهده و تغییر هستند. هر شی تحت مدیریت بر اساس یکی از این مدل‌های اطلاعاتی در سامانه توصیف می‌گردد. (FA07)

♦ عملیات‌هایی که بر روی هر شی قابل انجام است محدود به مواردی است که در مدل اطلاعاتی آن توصیف شده است. سرویس‌های مختلفی که توسط پلتفرم پیشنهادی ارائه می‌شود، یک نسخه از مدل اطلاعاتی اشیاء را در خود دارند و درخواست‌هایی که برای آن شی مد نظر توسط برنامه‌های کاربردی ارسال می‌شود، قبل از اعمال به شی بر اساس این مدل اطلاعاتی بررسی می‌گردند. (FA08)

♦ مدل اطلاعاتی اشیاء با یک زبان اختصاصی سامانه پیشنهادی توصیف می‌گردد که امکان صحت‌سنجی آن نیز بر اساس یک مدل مرجع وجود دارد. مدل‌های توصیف شده توسط این زبان در یک پایگاه‌داده ذخیره شده و به عنوان یکی از سرویس‌های پلتفرم پیشنهادی در اختیار برنامه‌های کاربردی و سایر سرویس‌های پلتفرم قرار می‌گیرد. (FA09)

♦ اطلاعات جمع‌آوری شده از اشیاء که در یک پایگاه‌داده ذخیره شده است توسط API مناسب در اختیار برنامه‌های کاربردی قرار می‌گیرد. برای این منظور امکان اتصال به پایگاه‌داده از طریق یک واسطه سطح بالاتر که دسترسی را کنترل کرده و جزییات پایگاه‌داده را از برنامه‌های کاربردی پنهان می‌سازد به عنوان یکی از سرویس‌های پایه سامانه پیشنهادی پیاده‌سازی می‌گردد. (FA10)

♦ علاوه بر امکان دسترسی به پایگاه‌داده برای برنامه‌های کاربردی، خود سامانه پیشنهادی نیز دارای واسطه کاربر گرافیکی برای نمایش داده‌ها در قالب نمودارهای مختلف است. انواع نمودارهای زمانی و جغرافیایی توسط این سرویس از سامانه پیشنهادی در اختیار مدیر سیستم قرار می‌گیرد. (FA11)

♦ با توجه حجم بالای داده تولید شده، استفاده از ابزارهای BIG DATA برای پردازش آنها لازم است. برای این منظور، سامانه پیشنهادی امکان اتصال این نوع ابزارهای پردازشی را به پایگاه‌داده پلتفرم فراهم می‌آورد. ولی با توجه به اینکه هر یک از این ابزارها دارای نیازمندی‌های خاص خود برای اتصال به پایگاه‌داده‌های مختلف هستند، جزییات نحوه ارتباط بعد از انتخاب نهایی پایگاه‌داده

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۲۰

و ابزار تحلیل مد نظر امکان‌پذیر خواهد بود. (FA12)

♦ از آنجایی که امکان افزودن پروتکل‌های ارتباطی خاص در سامانه وجود دارد و مدل اطلاعاتی اشیاء نیز در قالب یک زبان توصیف می‌گردد، فلذا سامانه پیشنهادی محدود فقط به یک نوع کاربرد خاص نبود و با سفارشی‌سازی و تغییر سرویس‌های آن (بدون نیاز به باز طراحی کلی) امکان استفاده از آن در کاربردهای متنوع وجود دارد. (FA13)

♦ API‌هایی که در اختیار برنامه‌های کاربردی قرار می‌گیرد از یک کانال کنترل دسترسی عبور می‌کند تا هر درخواست تنها به اشیای مجاز بتواند دسترسی داشته باشد. (FA14)

♦ یکی از مواردی که در کنترل دسترسی مورد بررسی قرار می‌گیرد وضعیت پرداخت و شارژ مشتری است. در صورت که شارژ کافی وجود نداشته باشد درخواست مربوطه با پیغام مناسب خاتمه می‌یابد. (FA15)

♦ سامانه پیشنهادی قابلیت اتصال به درگاه‌های پرداخت برای شارژ کاربران را دارد. این قابلیت به صورت یک سرویس در سامانه پیشنهادی پیاده‌سازی می‌گردد. (FA16)

• تبصره: با توجه به متفاوت بودن API درگاه‌های پرداخت آنلاین در کشور، جزییات این سرویس، بعد از تعیین تکلیف در این خصوص قابل استخراج است.

### ۳-۴- قابلیت‌های غیر کارکردی سامانه پیشنهادی

علاوه بر کارکردهای فوق‌الذکر که در سامانه پیشنهادی پیاده‌سازی می‌شوند، قابلیت‌های غیر کارکردی متنوعی که در درجه اهمیت بسیار زیادی در موفقیت راهکار مبتنی بر اینترنت اشیاء دارند، نیز در سامانه پیشنهادی پیاده‌سازی می‌شود که جزییات آنها در این بخش شرح داده شده است. مشابه قابلیت‌های کارکردی، هر یک از کارکردها دارای یک کد است.

♦ سامانه پیشنهادی دارای یک معماری توزیع شده است. قابلیت‌های کارکردی که در بخش قبل شرح داده شد در قالب سرویس‌های مجزا پیاده‌سازی می‌شوند. (NF01)

♦ با توجه به معماری توزیع شده پلتفرم پیشنهادی، این سامانه دارای قابلیت مقیاس‌پذیری افقی است که این امکان را فراهم می‌کند تا با افزایش ابعاد شبکه به راحتی بتوان با اضافه نمودن منابع سخت‌افزاری جدید پاسخ‌گوی نیازمندی‌ها بود. (NF02)

♦ با توجه به اینکه در معماری توزیع شده از یک سرویس می‌تواند چندین نمونه وجود داشته باشد،

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۲۱

بنابراین در سامانه پیشنهادی مکانیزم توزیع بار مناسبی تعبیه می شود تا از منابع پردازشی به صورت مناسب و بهینه استفاده گردد. (NF03)

♦ با توجه به اینکه در سامانه پیشنهادی هر کارکرد در قالب یک سرویس مجزا پیاده سازی می شود، بنابراین امکان فعال و غیر فعال نمودن هر یک از آنها به صورت مجزا (البته نه همه سرویس چرا که برخی از آنها برای کارکرد سیستم لازم است) وجود دارد. (NF04)

♦ به منظور تامین امنیت ارتباط بین اشیاء و پلتفرم، امکان استفاده از پروتکل های امنیتی در پشته پروتکلی ارتباط شی-پلتفرم در سامانه پیشنهادی وجود دارد. استفاده یا عدم استفاده از آنها نیاز به پیکربندی عامل هوشمندسازی در اشیاء و پلتفرم دارد. (NF05)

♦ به منظور تامین امنیت ارتباط بین برنامه های کاربردی و پلتفرم، امکان استفاده از پروتکل های امنیتی در پشته پروتکلی ارتباط پلتفرم-کاربرد در سامانه پیشنهادی وجود دارد. استفاده یا عدم استفاده از آنها نیاز به پیکربندی برنامه کاربردی و پلتفرم دارد. (NF06)

♦ به منظور حفظ امنیت داده، دسترسی به پایگاه داده توسط مکانیزم های کنترل دسترسی محدود می شود. (NF07)

♦ به منظور تامین امنیت خود پلتفرم، تنها پورتهایی که برای ارایه سرویس های پلتفرم لازم است بر روی سیستم عامل سامانه باز بوده و سایر پورها از بسته است. (NF08)

• **تبصره:** تامین امنیت شبکه ای که سامانه پلتفرم در آن نصب و راه اندازی می شود (مانند استفاده از Firewall و ...) خارج از حوزه این پیشنهاد است.

♦ به منظور کنترل دسترسی کاربران، به صورت پیش فرض سامانه پیشنهادی حداقل از دو نقش مدیر و کاربر سامانه پشتیبانی می کند که میزان دسترسی این دو نقش به ترتیب نوشتن/خواندن و تنها خواندن است. (NF09)

♦ امکان تعریف نقش برای کاربران و کنترل دسترسی آنها در محیط گرافیکی سامانه وجود دارد. (NF10)

♦ به منظور مقابله با خرابی پلتفرم، سامانه پیشنهادی دارای سرویسی است که سلامت سایر سرویس ها را بررسی و در صورت مشاهد مشکل اقدامات لازم را فراهم می نماید. در ضمن این قابلیت در کتابخانه ارایه شده برای توسعه عامل هوشمندسازی وجود دارد که اطلاعات را برای دو آدرس IP مختلف ارسال نماید. (NF11)

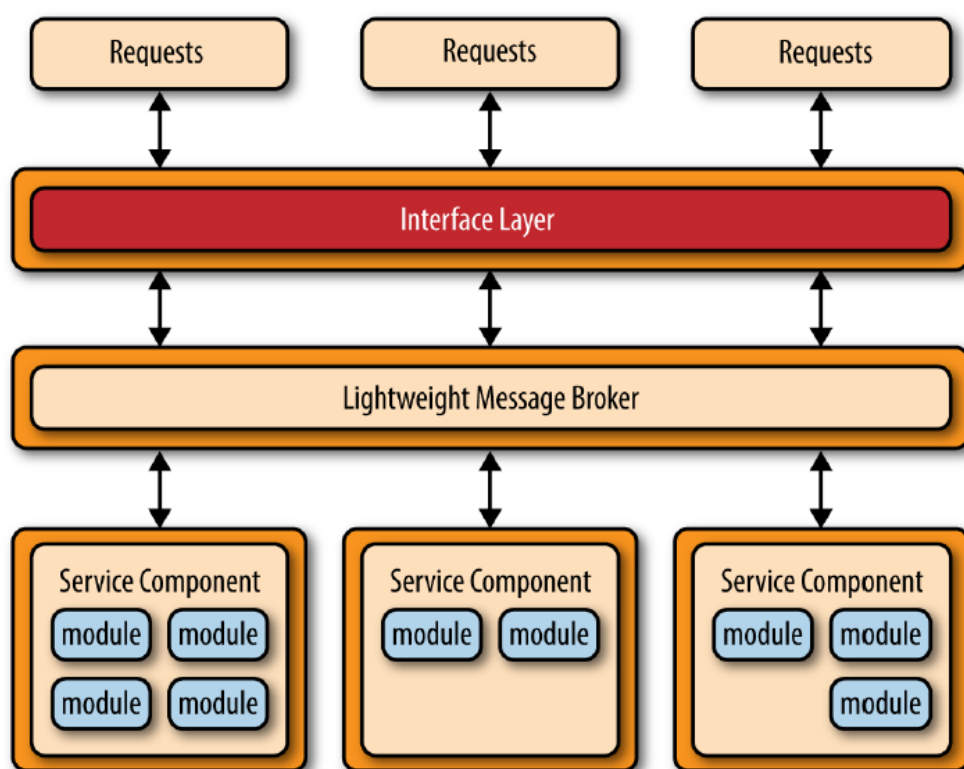
کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۲۲

### ۳-۵- معماری سامانه پیشنهادی

یکی از فاکتورهایی که نقش کلیدی در موفقیت یک محصول نرم‌افزاری دارد، معماری آن است. تامین همه نیازمندی‌ها بخصوص نیازمندی‌های غیرکارکردی در گرو استفاده از یک معماری مناسب برای سامانه است. معماری سامانه است که مشخص می‌کند این سامانه به چه نحوی به اجزای آن شکسته شده و این اجزا به چه نحوی در تعامل با یکدیگر هستند. در حوزه معماری سیستم‌های نرم‌افزاری، الگوهای معماری متعددی مانند Layered، Microkernel، Space-Based و Microservices وجود دارد.

#### ۳-۵-۱- معماری Microservices

معماری سامانه پیشنهادی مبتنی بر Microservices خواهد بود. ایده این معماری برخلاف روشهای سنتی سامانه‌های یکپارچه، یک سیستم توزیع شده متشکل از چندین سرویس است که هر سرویس به صورت مجزا پیاده‌سازی شده و سرویس مربوطه را از طریق API در اختیار سایر سرویس‌ها و مشتریان قرار می‌دهد. شمایی کلی این معماری در شکل (۳-۳) نشان داده شده است.



شکل (۳-۳) معماری Microservices

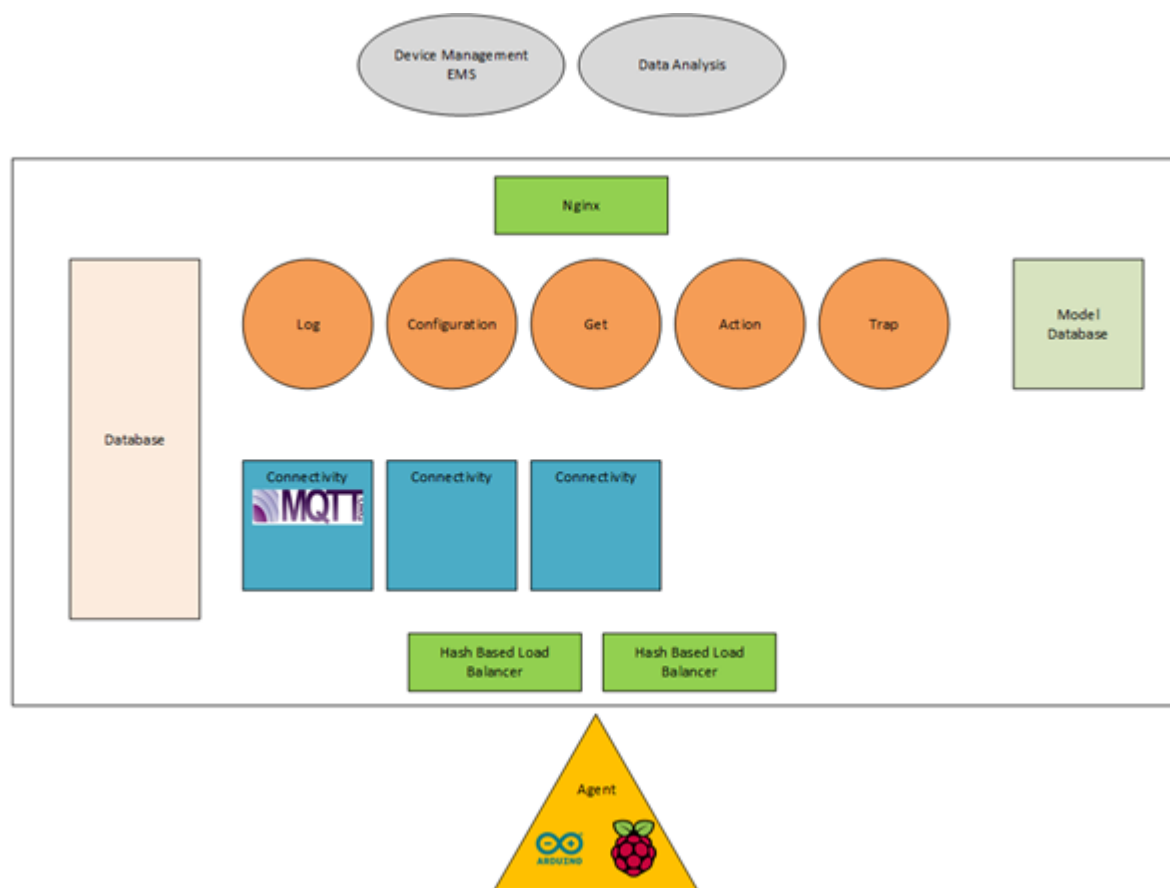
صفحه	تاریخ	کد سند
۲۳	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

در این معماری، کارکردهای سامانه در قالب سرویس‌هایی پیاده‌سازی می‌شوند که هر سرویس یک واحد قابل استقرار مستقل است. برای توزیع درخواست‌های داده شده به سامانه و همچنین ارتباطات بین سرویس‌ها عموماً از یک Message Broker در این معمار استفاده می‌شود.

استفاده از این معماری برای پیاده‌سازی پلتفرم بومی عام منظوره پیشنهادی دارای چندین مزیت است. اول اینکه اکثر نیازمندی‌های کارکردی سامانه می‌توانند به صورت سرویس‌های مجزا در قالب این معماری پیاده‌سازی شوند. این معماری به دلیل توزیع‌شدگی به صورت ذاتی مقیاس‌پذیر بوده و امکان اضافه کردن قابلیت‌های غیرکارکردی که در بخش پیش شرح داده شده به این معماری وجود دارد.

### ۳-۵-۲- معماری کلان پلتفرم پیشنهادی

معماری کلان این سامانه بر اساس معماری Microservices و قابلیت‌های در شکل (۳-۴) آمده است.



شکل (۳-۴) معماری کلان سامانه پلتفرم پیشنهادی

در این معماری هر کارکرد توسط یک سرویس پیاده‌سازی می‌شود که سرویس‌های اصلی در این شکل

صفحه	تاریخ	کد سند
۲۴	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

نشان داده شد است. در پایین ترین سطح سرویس Connectivity وجود دارد که پروتکل های ارتباط با اشیاء مانند MQTT را پیاده سازی می کند. بعد از آن سرویس های پایه وجود دارند که قابلیت های گزارش دوره ای، set، get و .... را پیاده سازی می کنند. این سرویس ها از سرویس های DB و Model برای ذخیره داده و بررسی مدل اطلاعاتی استفاده می کنند. در سطح بالاتر سرویس هایی مانند Device Management وجود دارد که از سرویس های پایه استفاده می کند. برای تامین مقیاس پذیری امکان تقسیم بار و load balancing هم در ارتباط با اشیاء و هم در ارتباط با برنامه های کاربردی در این معماری دیده شده است.

### ۳-۶- متدولوژی توسعه

فاکتور دوم تعیین کننده در موفقیت پروژه های نرم افزاری، متدولوژی توسعه نرم افزار است که چارچوب مدیریت فرآیند توسعه سامانه است. این چارچوب مشخص می کند که توسعه سامانه طی چه مراحل باید انجام گیرد. استفاده از چارچوب مناسب و متناسب با ماهیت سامانه مد نظر تاثیر مستقیمی بر کیفیت و هزینه های زمانی و مالی محصول خواهد داشت. برای توسعه نرم افزار، متدولوژی های مختلفی مانند Agile، Crystal Methods، Extreme Programming، Rapid Application Development، Scrum و Waterfall در طول زمان ابداع شده است.

نکته حائز اهمیت در این خصوص این مساله است برخی از این معماری ها با برخی از متدولوژی های توسعه همخوانی بیشتر دارند. بنابراین اگر به ترکیب درست این دو توجه شود، فرآیند توسعه کارایی بالایی داشته و کیفیت محصول افزایش خواهد یافت. در این پروژه برای توسعه سامانه پیشنهادی از متدولوژی Scrum در کنار معماری Microservices استفاده خواهد شد.

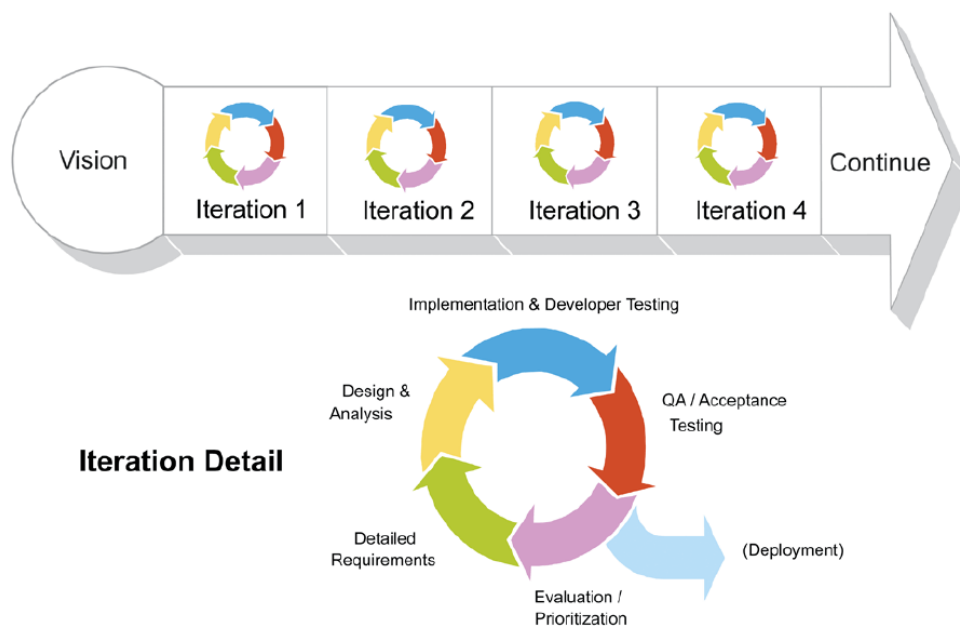
### ۳-۶-۱- متدولوژی Scrum

متدولوژی (چارچوب) Scrum که جزء دسته چارچوب های Agile به شمار می آید بر این فرض استوار است که الزاما همه نیازمندی های سامانه در ابتدای کار به صورت کاملا شفاف و ثابت بیان نشده، بلکه این نیازمندی ها در طول زمان مشخص شده و تغییر می باید. در چارچوب Scrum، توسعه سامانه به صورت افزایشی (incremental) در قالب زیرپروژه هایی که به اسم Sprint شناخته می شود توسط تیم های خودسازمانده انجام

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۲۵



دهد. Sprint ها یک بخش از محصول هستند که حداکثر طی ۱ ماه توسط یک تیم حداکثر ۶ نفره قابل انجام و ارایه است. در این چارچوب، بخش های مختلف سامانه در قالب Sprint تعریف شده و هر یک از آنها در یک چرخه نیازسنجی، طراحی، پیاده سازی و تست می شوند. فرآیند کلی این چارچوب در شکل زیر نشان داده شده است.



شکل (۳-۵) متدولوژی Scrum

چارچوب Scrum برای توسعه پلتفرم عام منظوره پیشنهادی استفاده خواهد شد چرا که اولاً، چارچوب Scrum که جزء متدولوژی های Agile است بر روی توسعه سریع و چابک محصول تمرکز دارد که با زمان بندی این پروژه کاملاً سازگار است. ثانیاً، هر یک از نیازمندی های سامانه پیشنهادی که در بخش های پیشین شرح داده شد، قابل انجام در قالب Sprint های این چارچوب است چرا که حجم کاری آنها متناسب با حجم کاری در نظر گرفته شده با Sprint ها در این متدولوژی است. ثالثاً، با گذر زمان، نیاز به سفارشی سازی این پلتفرم برای کاربردهای مختلف وجود خواهد داشت که با ایده اصلی این چارچوب که فرآیند توسعه را یک فرآیند تکرار شونده در نظر می گیرد، همخوانی دارد.

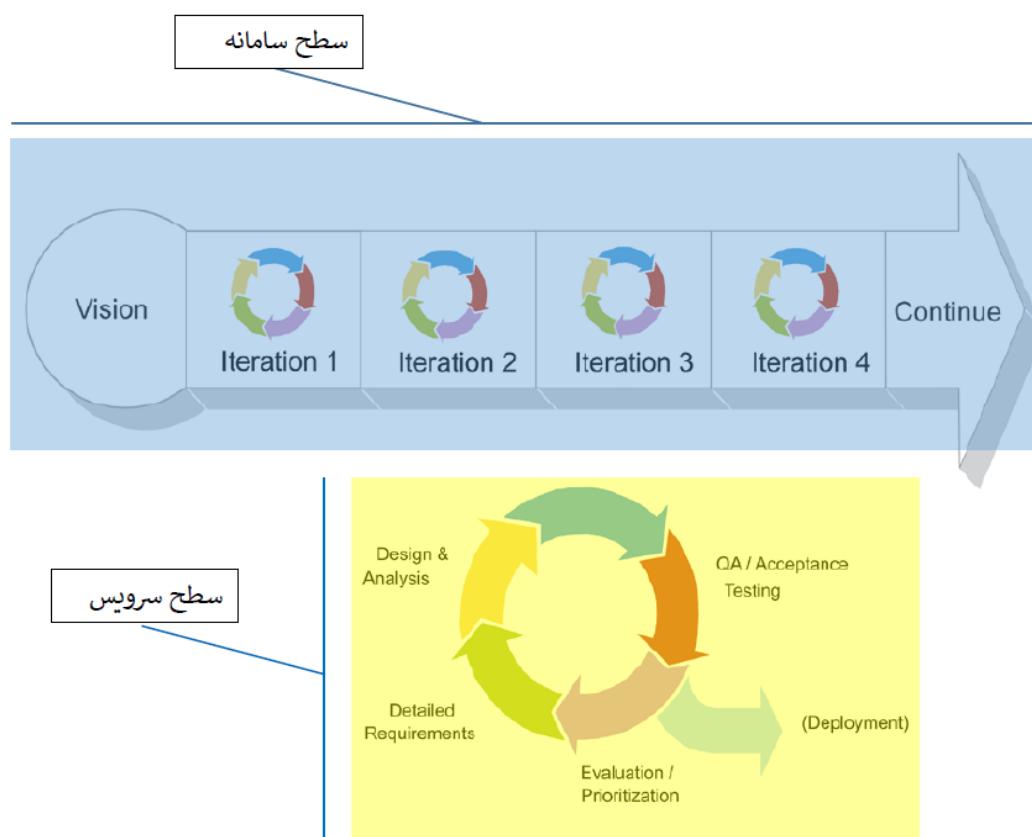
نکته آخری که به عنوان مزایای استفاده از Scrum و Microservices برای توسعه سامانه پیشنهادی باید آن اشاره کرد، سازگاری مناسب این متدولوژی و معماری با یکدیگر است. هر یک از سرویس های این معماری با تعریف کلی که از کارکرد آن وجود دارد میتواند در قالب یک Sprint در چارچوب Scrum

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۲۶

پیاده‌سازی شده و مستقر گردد. هر نیاز به تغییر در قابلیت‌های و جزئیات یک سرویس نیز مستقل از سایر سرویس‌ها می‌تواند به عنوان یک Sprint جدید صورت پذیرد.

### ۳-۶-۲- فرآیندهای انجام پروژه

بر اساس استفاده از متدولوژی Scrum و معماری Microservices، طراحی، پیاده‌سازی، تست و مستندسازی سامانه در قالب دو دسته فرآیند کلی الف) سطح سامانه و ب) سطح سرویس انجام می‌شود. همانگونه که در شکل زیر نشان داده شده است، فرآیندهای سطح سامانه مجموعه فرآیندهایی هستند که کلیت توسعه سامانه را بدون وارد شدن به جزئیات فنی انجام می‌دهند و فرآیندهای سطح سرویس دقیقاً معادل فرآیندهایی است که در چارچوب Scrum برای یک Sprint در نظر گرفته شده است.



شکل (۳-۶) فرآیندهای پروژه در سطح سیستم و سرویس

### ۳-۶-۲-۲- فرآیندهای سطح سامانه

این دسته از فرآیندها به صورت کلان به طراحی سامانه، تعیین سرویس‌ها، الویت‌بندی و تجمیع سرویس‌ها

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۲۷

می‌پردازد که به صورت جزئی تر به شرح زیر است.

- ♦ **طراحی کلان:** براساس نیازمندی‌های مد نظر برای سامانه، سرویس‌های مد نظر برای تامین نیازمندی‌های کارکردی و غیرکارکردی استخراج می‌شود که در آن مشخص است هر سرویس چه کارکردی را پیاده‌سازی می‌کند.
- ♦ **تعریف Sprint:** بر اساس طراحی کلان، Sprintها تعریف شده و انجام آنها الویت‌بندی می‌شود.
- ♦ **تجمیع:** سرویس‌های توسعه داده شده در قالب سیستم نهایی تجمیع شده و مورد تست و ارزیابی قرار گیرند.
- ♦ **استخراج چارچوب فرآیندهای سطح سرویس:** در سطح سرویس فرآیندهای متعددی برای پیاده‌سازی هر سرویس انجام می‌گیرد. چارچوب این فرآیند و نحوه مستندسازی آنها به عنوان یکی از وظایف سطح سیستم است.

### ۳-۶-۲-۳- فرآیندهای سطح سرویس

هر Sprint که معادل طراحی و توسعه یک سرویس است از فرآیندهای ذیل تشکیل شده است.

- ♦ **تحلیل نیازمندی‌ها:** در این فرآیند، نیازمندی‌های مطرح شده برای این سرویس که به صورت کلی است، مورد تحلیل قرار گرفته و نیازمندی‌های جزئی فنی آن استخراج می‌گردد. علاوه بر نیازمندی‌های کارکردی، یکی از مواردی که در این فرآیند مشخص می‌شود API این سرویس است.
- ♦ **طراحی:** در این فرآیند، طراحی داخلی سرویس مد نظر انجام می‌شود. علاوه بر آن تکنولوژی‌های مد نظر برای پیاده‌سازی این سرویس نیز بررسی شده و گزینه مناسب انتخاب می‌گردد.
- ♦ **پیاده‌سازی:** در این فرآیند، پیاده‌سازی سرویس مد نظر با استفاده از تکنولوژی انتخاب شده انجام می‌گیرد.
- ♦ **تست:** در این مرحله از Sprint، ابتدا سناریوهای تست این سرویس تهیه می‌شود که عموماً تست‌های از نوع Unit Test و Component Interface Test و به صورت Black-box (و در صورت نیاز White-Box) است. سپس، تست این سرویس بر اساس سناریوهای مذکور انجام گرفته و مشکلات احتمالی رفع می‌گردد.
- ♦ **استقرار:** با توجه به اینکه در معماری Microservices هر سرویس یک واحد مستقل استقرارپذیر است، در این مرحله استقرار سرویس مد نظر انجام می‌پذیرد.

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۲۸

♦ **مستند سازی:** به موازات فرآیندهای فوق الذکر، فرآیند مستند سازی این سرویس انجام می شود که شامل الف) جزییات نیازمندی های سرویس، ب) جزییات طراحی سرویس، ج) سناریوهای تست و نتایج آنها و د) نحوه استقرار سرویس است.

### ۷-۳- تخمین منابع سخت افزاری لازم برای میزبانی پلتفرم

تخمین منابع سخت افزاری لازم برای میزبانی پلتفرم پیشنهادی برای ۵۰۰۰ نود با قابلیت امکان مقابله با خرابی سخت افزاری سرور در جدول (۱-۳) نشان داده شده است.

جدول (۱-۳) تخمین سخت افزار مورد نیاز

منابع			سرویس	ردیف
Storage (GB)	RAM (GB)	CPU (core)		
500	32	8	DB + Data Analysis	۱
600	56	14	Get, Set, Push, Trap, Model, Connectivity, API, UI, Fault, Configuration, Accounting, Performance, Security, LB	۲
400	40	10	Other Services	۳
1500	128	32	مجموع منابع سرور اصلی	
1500	128	32	مجموع منابع سرور پشتیبان	

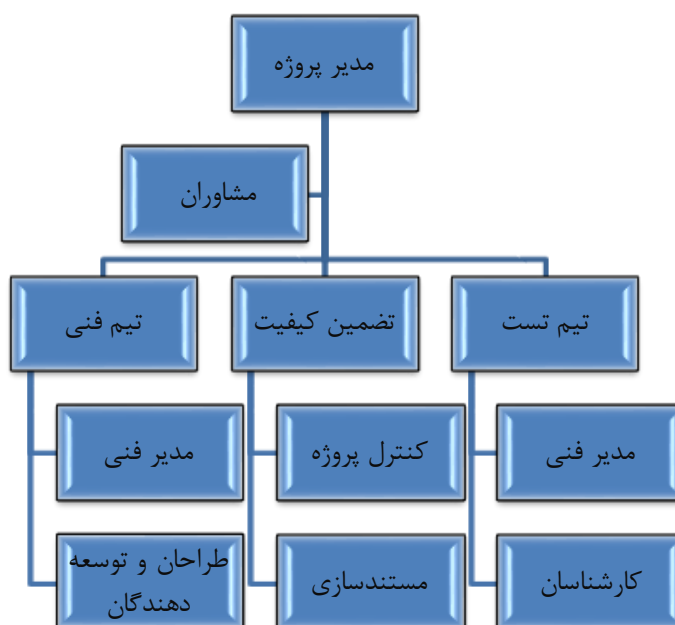
## فصل ۴:

# ساختار سازمانی

صفحه	تاریخ	کد سند
۳۰	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## ۴-۱- چارت سازمانی انجام پروژه

تیم پیشنهاد دهنده برای انجام این پروژه در شکل زیر نشان داده شده است که در آن سه زیر تیم فنی، تضمین کیفیت و تست تحت نظر مدیر پروژه که از مشاوران استفاده می کند، پروژه را انجام می دهند. افراد کلیدی این ساختار سازمانی در جدول (۴-۱) ذکر شده که شرح سوابق و رزومه آنها در پیوست ۱ آمده و جزئیات شرح وظایف این واحدها در بخش بعدی شرح داده شده است.



شکل (۴-۱) چارت سازمانی انجام پروژه

جدول (۴-۱) افراد کلیدی تیم پیشنهاد دهنده

ردیف	مسئولیت	مسئول	تحصیلات
۱	مدیر پروژه	بهادر بخشی	دکتری
۲	مشاوران	مهدی صدیقی امین قاسم صفریان سعید سعیدی	دکتری
۳	مدیر فنی	بهادر بخشی	دکتری
۴	مدیر تضمین کیفیت	مسعود صبایی	دکتری
۵	مدیر تست	مهدی راستی	دکتری
۶	کارشناس طراح	پرهام الوانی	دانشجوی کارشناسی ارشد
۷	کارشناس تست	روژین اصلانی	دانشجوی دکتری
۸	کارشناس تضمین کیفیت	عاطفه ترمچی	دانشجوی دکتری

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۳۱

چند نکته در خصوص اعضای تیم پیشنهاد دهنده قابل ذکر است:

- ♦ جدول (۴-۱) تنها افراد کلیدی تیم را نشان می‌دهد و واضح است که تیم پیشنهاد دهنده شامل نیروهای بیشتری خواهد بود که در فاز اول پروژه نهایی می‌گردد.
- ♦ اعضای تیم پیشنهاد دهنده به سه سطح ۱ و ۲ و ۳ دسته‌بندی می‌گردد. مدیر پروژه، مدیران تیم‌ها و مشاوران در سطح ۱ قرار دارند، کارشناسان ارشد در سطح ۲ قرار دارند و سطح ۳ کارشناسان را شامل می‌شود. تخصص‌های مورد نیاز و نرخ پرداختی آنها در مستند تحلیل هزینه ارایه گردیده است.
- ♦ به علت ماهیت تیم پیشنهادی (تیم دانشگاهی متشکل از اساتید، دانشجویان و فارغ التحصیلان)، اعضای آن به صورت تمام وقت در پروژه حضور ندارد. بلکه میزان مشارکت آنها بر اساس حجم کار فازهای پروژه تغییر می‌یابد که جزییات آن در مستند تحلیل هزینه شرح داده شده است.
- ♦ دو نیروی مستقر در محل کارفرما، از کارشناسان سطح ۲ جناب آقایان الوانی و برقی هستند.

## ۴-۲- وظایف واحدها

در چارت سازمانی نشان داده شده در شکل (۴-۱)، هر یک از واحدها، نقش و وظایف تعریف شده به شرح زیر را در پروژه بر عهده دارند.

♦ **مدیر پروژه:** مسئولیت کل پروژه بر عهده مدیر پروژه است و وظایف زیر را بر عهده دارد:

- نظارت و تایید طراحی کلان سامانه
- استخراج Sprintها، الویت‌دهی و نظارت بر اجرای آنها
- برگزاری جلسات کنترل پروژه
- تصمیم‌گیری در خصوص اعضای تیم
- هماهنگی بین تیم‌های توسعه فنی و تست و کنترل

♦ **مدیر فنی تیم توسعه:** در چارچوب Scrum به اسم «Scrum Master» شناخته می‌شود و مسئول موارد زیر است:

- مدیریت فنی تیم‌های طراحی و توسعه
- نظارت بر طراحی و پیاده‌سازی

صفحه	تاریخ	کد سند
۳۲	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

- تعامل با اعضای تیم برای انطباق با مدل Scrum
- حل مشکلات تعامل فی مابین تیم‌ها
- افزایش توانایی تیم فنی
- هماهنگی تعاملات مدیر پروژه با اعضای تیم‌های فنی
- ارائه گزارش‌های پیشرفت به مدیر پروژه

♦ طراحان و توسعه دهندگان: این افراد به چندین گروه حداکثر ۶ نفره تقسیم شده و مسئول انجام یک Sprint هستند. وظایف اصلی آنها عبارت است از:

- انتخاب، برنامه‌ریزی و زمانبندی توسعه یک سرویس در تعامل با مدیران
- طراحی فنی سرویس‌ها
- پیاده‌سازی سرویس‌ها
- استقرار و تست سرویس‌ها
- مستندسازی توسعه
- ارائه گزارش به مدیر فنی

♦ مدیر تست

- زمانبندی و الویت‌بندی تست‌ها در تعامل با مدیر پروژه
- مدیریت و نظارت بر تهیه سناریوهای تست
- مدیریت و نظارت بر راه‌اندازی محیط و ابزارهای تست
- مدیریت برگزاری تست‌ها
- تعامل با مدیر فنی برای رفع مشکلات
- ارائه گزارش به مدیر پروژه

♦ کارشناسان تست

- تدوین سناریوهای تست
- راه‌اندازی ابزارها و محیط تست
- برگزاری تست و مستندسازی و گزارش خطاها

صفحه	تاریخ	کد سند
۳۳	۹۶/۵/۸	IoT-PF-PM-0010-v1.00



- ارایه گزارش به مدیر فنی تست

♦ کنترل پروژه

- تشکیل جلسات مستمر کنترل پروژه مطابق با استاندارد PMBOK
- کنترل و پیگیری امور محوله به کلیه تیم‌ها
- کنترل برنامه‌های زمانبندی پروژه
- تهیه و ارسال گزارشات مشکلات به مدیر پروژه و پیگیری آنها
- تهیه گزارشات هفتگی و ماهیانه

♦ مستندسازی

- تهیه قالب‌های گزارشهای فنی
- تهیه قالب موارد آموزشی
- پیگیری و نظارت بر مستندسازی کد
- بازبینی مستندات
- تهیه گزارشها و ارایه برای کارفرما

صفحه	تاریخ	کد سند
۳۴	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## فصل ۵:

# روش مدیریت و تضمین کیفیت

صفحه	تاریخ	کد سند
۳۵	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## ۵-۱- کلیات طرح مدیریت پروژه

در مدیریت پروژه، از استاندارد PMBOK استفاده می شود که با توجه به زمان محدود پروژه، حوزه های مدیریت پیکربندی، مدیریت زمان، مدیریت کیفیت، مدیریت منابع و مدیریت ریسک ها از اهمیت بالایی برخوردار است. ابزاری که برای مدیریت پروژه استفاده خواهد شد نرم افزار متن باز OpenProject است که قابلیت های مختلفی مانند تعریف زمان بندی، اختصاص وظیفه (task)، تنظیم جلسات و موارد مشابه دیگر را دارد و با متدولوژی Scrum هم مطابقت کامل دارد. برای مدیریت پیکربندی و نسخه بندی کدها نیز از بسته نرم افزاری متن باز GitLab استفاده می شود.

برای اجرای موفقیت آمیز پروژه و تضمین کیفیت آن، در ابتدای پروژه سندی تحت عنوان طرح مدیریت پروژه (Project Management Plan, PMP) تهیه شده و به تأیید کارفرما خواهد رسید. در این سند کلیه موارد لازم برای جهت مدیریت و تضمین کیفیت و نحوه تعامل بین پیمانکار و کارفرما ارائه خواهد شد. در ادامه کلیات طرح های مدیریت پیکربندی، تضمین کیفیت و مدیریت ریسک ها که از جمله بخش های مهم، سند PMP است شرح داده می شود.

## ۵-۲- کلیات طرح مدیریت پیکربندی

جهت ارتباط منطقی بین مستندات پروژه روش شماره گذاری مناسبی برای مستندات طراحی خواهد شد که قابلیت ردیابی، خروجی های (محصولات) تولید شده در هر فاز سیستم در آن مشخص شده است و با ارائه یک روش سلسله مراتبی شماره گذاری امکان کنترل و مدیریت کلیه مستندات وجود خواهد داشت.

برای مدیریت تغییرات و پیکربندی کدهای تولید شده از سیستم مدیریت تغییرات GitLab استفاده خواهد شد تا به امکان نسخه بندی و مدیریت کدها وجود داشته باشد.

## ۵-۳- کلیات طرح تضمین کیفیت

کنترل کیفیت تمامی خروجی های پروژه از لحاظ محتوایی، تطبیق با استانداردها و از نظر فنی، توسط مسئولین ذیربط مورد بررسی قرار می گیرد. تمامی مستندات و خروجی های تولید شده توسط تیم کنترل

صفحه	تاریخ	کد سند
۳۶	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

کیفیت ابتدا از نظر محتوایی مورد بررسی قرار می گیرند و در صورت تأیید، از نظر تطابق با استانداردها مربوطه بررسی خواهند شد و سپس مسئول فنی پروژه کنترل فنی مستندات و خروجی ها را انجام خواهد داد.

در بررسی کیفیت پروژه، فرم هایی که به منظور تضمین کیفیت طراحی خواهند شد، توسط کارشناسان تیم پروژه استفاده خواهد شد و مدیر تضمین کیفیت پروژه پس بررسی های انجام شده، موارد نیاز به اصلاح را برای اصلاح به تیم پروژه ارجاع خواهد داد. نتیجه بازبینی ممکن است قبول، قبول مشروط، مردود و یا معوق باشد که در هر مورد بنا به شرایط اقدام خواهد شد.

تیم تضمین کیفیت خلاصه و نتیجه گزارش مشکلات را به مدیر پروژه، ناظر و مسئولین ذیربط ارسال می کند و در گزارشات هفتگی کنترل پروژه آمار و اطلاعات وضعیت کیفی خروجی ها گزارش خواهد شد.

## ۴-۵- کلیات طرح مدیریت ریسک

مشکلات و مخاطرات پروژه از پیش در طرح مدیریت پروژه نهایی فهرست گردیده و پس از تایید چارچوب شرح خدمات و نهایتاً راهکارهای پیشنهادی مدیریت پروژه در هر مورد ارائه می گردد. لیکن ساختار پایین به بالای این روش به عنوان شیوه اجرایی کار به این صورت است که تیم کارشناسی موظف است مشکلات مواجه شده را هر روز در گزارش روزانه و در خاتمه هفته در گزارش هفتگی عرضه نموده، مشکلات قابل حل توسط مدیر پروژه حل گردیده و سایر مشکلات به صورت رسمی به کارفرما ارائه خواهد شد. در مورد ریسک های پروژه به صورت مشابه بنا به بخش راهکارهای پیشنهادی عمل گردیده و ریسک ها به مدیران کارفرما اعلام خواهد شد.

۱. ارائه لیست مشکلات و مخاطرات پروژه به صورت روزانه توسط کارشناسان (رویه داخلی)

۲. جمع بندی موارد در گزارش هفتگی (رویه داخلی)

۳. بررسی موارد در جلسات هفتگی توسط مدیر پروژه (رویه داخلی)

۴. حل و فصل مورد توسط مدیر پروژه (رویه داخلی)

۵. اعلام کتبی موارد غیرقابل رفع به کارفرما.

صفحه	تاریخ	کد سند
۳۷	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## ۵-۵- کلیات طرح مستندسازی

مستندسازی این پروژه در چندین سطح انجام می‌شود. یک دست از مستندات، مستندات مربوط به طراحی کلان و سیستمی سامانه است که بدون وارد شدن به جزئیات اجزای معماری ساختار کلان را مستندسازی می‌کند. در سطح بعدی مستندات مربوطه به اجزای معماری است که برای هر جزء مستندات مربوط به طراحی و پیاده‌سازی و در نهایت تست آن انجام می‌شود. سطح سوم مستندات، مستنداتی هستند که از کامنت‌ها و توضیحات کدها استخراج می‌شود. برای این منظور از بسته نرم‌افزاری doxygen استفاده خواهد شد. این بسته نرم‌افزاری امکان ایجاد مستندات تحت وب (و سایر فرمت‌ها) را از داخل کامنت‌های کد دارا است. لازم به ذکر است که ساختار همه این مستندات در فاز اول (که در بخش بعدی شرح داده شده است) در قالب سند توسعه سیستم نهایی خواهد شد.

به منظور تضمین کیفیت سامانه نهایی، همه روال‌های مربوط به جریان‌های کاری، فرآیندهای طراحی، فرآیندهای تست و مستندسازی در قالب سند توسعه سیستم استخراج شده و در اختیار تیم فنی قرار می‌گیرد تا همه فرآیندهای پروژه بر اساس رویه‌های استاندارد و مدون انجام گیرد. کنترل و نظارت انجام فرآیندها طبق رویه‌های مذکور از جمله وظایف مدیر فنی و تیم کنترل پروژه است.

صفحه	تاریخ	کد سند
۳۸	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## فصل ۶:

# فازبندی و شرح خدمات

صفحه	تاریخ	کد سند
۳۹	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## ۶-۱- فازبندی

طراحی و توسعه سامانه پلتفرم پیشنهادی در طی چهار فاز اصلی به شرح زیر انجام می گیرد که جزئیات خدمات ارایه شده در هر فاز در بخش آتی شرح داده شده است.

### ♦ فاز ۱: ایجاد زیرساخت انجام پروژه

هدف از این فاز ایجاد زیرساختهای لازم برای انجام پروژه است. این اقدامات زیرساختی شامل الف) ایجاد تیم پروژه، ب) راه اندازی مکانیزمهای مدیریت و کنترل پروژه، ج) تعیین متدولوژی توسعه و تبیین آن برای تیم و د) تدوین فرآیندهای سطح سامانه و سطح سرویس است. علاوه بر آن قالب مستندات و خروجی های فازهای بعدی نیز در سند توسعه سیستم نهایی می گردد.

### ♦ فاز ۲: طراحی سطح سیستم

هدف این فاز طراحی کلان سامانه و استخراج سرویس های مد نظر و به تبع آن مشخص کردن Sprint ها است. با وجود اینکه قسمت عمده این فاز در ابتدای پروژه انجام می شود ولی این فاز تا انتهای پروژه همچنان باز است تا در صورت نیاز Sprint های جدید تعریف گردد.

### ♦ فاز ۳: طراحی و توسعه سرویس ها

این فاز یک فاز تکرار شونده است و برای همه Sprint ها، فرآیندهای تشریح شده در سطح سرویس را اجرا می کند. بنابراین هدف اصلی این فاز طراحی، توسعه، تست، استقرار و مستندسازی هریک از سرویس های مد نظر در سامانه است.

### ♦ فاز ۴: جمعیت و تست میدانی

این فاز با هدف جمعیت همه سرویس های سامانه و تست میدانی آنها در نظر گرفته شده است تا از پوشش نیازمندی های RFP توسط سامانه توسعه داده شده اطمینان حاصل گردد.

### ♦ فاز ۵: تحویل و انتقال دانش

در این فاز، به عنوان آخرین فاز پروژه، هدف ارایه سامانه توسعه داده شده و جزئیات آن به کارفرما در قالب مستندات و کلاس های آموزشی است.

## ۶-۲- شرح خدمات و خروجی ها

طبق فازبندی فوق الذکر، خدمات ذیل در هر فاز ارایه شده خروجی مشخص شده تحویل داده خواهد

شد.

صفحه	تاریخ	کد سند
۴۰	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

### ♦ فاز ۱: ایجاد زیرساخت انجام پروژه

- (۱-۱) نهایی سازی متدولوژی توسعه سامانه
  - (۲-۱) تشکیل تیم و مطابقت آن با متدولوژی انتخاب شده
  - (۳-۱) راه اندازی سیستم مدیریت و کنترل پروژه
  - (۴-۱) استخراج و تدوین فرآیندهای سطح سامانه و سطح سرویس
  - (۵-۱) تدوین سند توسعه سیستم
- خروجی:** مستندات مدیریت پروژه و توسعه سیستم

### ♦ فاز ۲: طراحی سطح سیستم

- (۱-۲) استخراج لیستی از پلتفرم های رایج
  - (۲-۲) بررسی و مقایسه پلتفرم ها و شکست الزامات مطرح شده در RFP به قابلیت های سیستم های موجود
  - (۳-۲) انتخاب الگوی معماری سامانه مبتنی بر Microservices
  - (۴-۲) تعیین تکنولوژی و پروتکل های لازم برای پیاده سازی زیرساخت Microservices
  - (۵-۲) تضمین پوشش نیازمندی های غیر کارکردی سامانه در سطح معماری کلان
  - (۱-۵-۲) استخراج طرح امنیت سامانه
  - (۲-۵-۲) استخراج طرح مقیاس پذیری سامانه
  - (۳-۵-۲) استخراج طرح High Availability سامانه
  - (۴-۵-۲) استخراج سرویس های مد نظر
  - (۶-۲) نگاشت نیازمندی های کارکردی به سرویس جهت تضمین پوشش نیازمندی ها
  - (۷-۲) تعریف Sprint و نظارت بر اجرای آنها
- خروجی:** سند طراحی کلان سامانه

### ♦ فاز ۳: طراحی و توسعه سرویس ها

- (۱-۳) انجام مراحل زیر برای هر یک از Sprint ها
  - تحلیل نیازمندی: استخراج نیازمندی های فنی سرویس
  - طراحی: طراحی داخلی سامانه و واسطه های ارتباط آن و انتخاب تکنولوژی

صفحه	تاریخ	کد سند
۴۱	۹۶/۵/۸	IoT-PF-PM-0010-v1.00



- پیاده‌سازی
- تست: استخراج فرایند تست هر مولفه (unit test) و تست بر اساس آن
- استقرار
- مستندسازی

**تبصره:** فرآیندهای فوق‌الذکر برای سرویسهای Connectivity، Push، Pull،

Database، UI، Device Management که از هم اکنون محضر است و سایر

سرویس‌هایی که در فاز دوم استخراج می‌شود تکرار می‌گردد.

**خروجی:** سرویسهای توسعه داده شده به همراه مستندات آنها

#### ♦ فاز ۴: تجمیع و تست میدانی

۴-۱) تجمیع سرویس‌ها و استقرار سامانه برای تست

۴-۲) تدوین سناریوهای تست

۴-۳) آماده‌سازی محیط و ابزارهای تست

۴-۴) انجام تست‌ها و مستندسازی آن

**خروجی:** گزارش تست‌های پوشش دهنده RFP

#### ♦ فاز ۵: تحویل و انتقال دانش

۵-۱) استقرار نهایی سامانه در محل کارفرما

۵-۲) آماده‌سازی منابع آموزشی

۵-۳) برگزاری دوره آموزشی

**خروجی:** مستندات و کلاس‌های آموزشی

صفحه	تاریخ	کد سند
۴۲	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

## فصل ۷: زمان بندی

صفحه	تاریخ	کد سند
۴۳	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

پروژه پیشنهادی در طی مدت ۴ ماه (۱۷ هفته) طبق زمان‌بندی نشان داده شده در جدول (۷-۱) انجام خواهد شد. لازم به ذکر است شروع این زمان‌بندی از زمان پرداخت پیش پرداخت بعد از انعقاد قرارداد است.

جدول (۷-۱) زمان‌بندی انجام پروژه

فاز	فعالیت	زمان (هفته)																
		۱۷	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱
فاز ۱: ایجاد زیرساخت انجام پروژه	نهایی سازی متدولوژی توسعه سامانه																*	
	تشکیل تیم															*	*	*
	راه اندازی سیستم مدیریت پروژه															*	*	*
	استخراج و تدوین فرآیندهای سطح سامانه و سطح سرویس															*	*	*
	استخراج سند توسعه سیستم															*	*	*
فاز ۲: طراحی سطح سیستم	استخراج لیستی از پلتفرم‌های رایج															*		
	بررسی و مقایسه پلتفرم‌ها													*	*	*		
	انتخاب الگوی معماری سامانه مبتنی بر Microservices													*	*	*		
	تعیین تکنولوژی و پروتکل‌های لازم برای پیاده‌سازی زیرساخت Microservices													*	*	*		
	استخراج طرح امنیت سامانه													*	*	*		
	استخراج طرح مقیاس‌پذیری سامانه													*	*	*		
	استخراج طرح High Availability سامانه													*	*	*		
	استخراج سرویس‌های مد نظر													*	*	*		
	نگاشت نیازمندی‌های کارکردی به سرویس جهت تضمین پوشش نیازمندی‌ها									*	*	*	*	*	*	*		
	تعریف Sprint و نظارت بر اجرای آنها		*	*	*	*	*	*	*	*	*	*	*	*	*	*		
فاز ۳: پیاده‌سازی سرویسها	سرویس مدل اطلاعاتی													*	*			
	سرویس connectivity													*	*			
	سرویس periodic push													*	*			
	سرویس get													*	*			
	سرویس set													*	*			
	سرویس Database								*	*	*	*	*	*	*			
	سرویس trap								*	*	*	*	*	*	*			
	سرویس API								*	*	*	*	*	*	*			
	سرویس Front-End برای User Interface							*	*	*	*	*	*	*	*			
	سرویس Back-End برای User Interface							*	*	*	*	*	*	*	*			
	سرویس register								*	*	*	*	*	*	*			
	سرویس مدیریت پیکره بندی							*	*	*	*	*	*	*	*			
	سرویس مدیریت حسابرسی							*	*	*	*	*	*	*	*			
	سرویس مدیریت کارایی							*	*	*	*	*	*	*	*			
	سرویس مدیریت امنیت							*	*	*	*	*	*	*	*			
	سرویس مدیریت خرابی							*	*	*	*	*	*	*	*			
فاز ۴: جمعیت و تست میدانی	سایر سرویسها				*	*	*	*	*	*	*	*	*	*	*			
	جمعیت سرویسها و استقرار سامانه برای تست			*	*	*	*	*	*	*	*	*	*	*	*			
	تدوین سناریوهای تست			*	*	*	*	*	*	*	*	*	*	*	*			
	آماده‌سازی محیط و ابزارهای تست			*	*	*	*	*	*	*	*	*	*	*	*			
فاز ۵: تحویل و انتقال دانش	تست و مستندسازی		*	*	*	*	*	*	*	*	*	*	*	*	*			
	استقرار نهایی سامانه در محل کارفرما	*	*	*	*	*	*	*	*	*	*	*	*	*	*			
	آماده‌سازی منابع آموزشی	*	*	*	*	*	*	*	*	*	*	*	*	*	*			
فاز ۶: برگزاری دوره آموزشی	برگزاری دوره آموزشی	*	*	*	*	*	*	*	*	*	*	*	*	*	*			
		*	*	*	*	*	*	*	*	*	*	*	*	*	*			

## فصل ۸:

# پوشش RFP

صفحه	تاریخ	کد سند
۴۵	۹۶/۵/۸	IoT-PF-PM-0010-v1.00

این پیشنهاد پروژه، همه بندهای RFP را پوشش داده است که جزییات آن در جدول (۸-۱) آمده است. در این جدول ستون «صفحه RFP» به همراه «موضوع بند RFP»، یکی از نیازمندی‌های مطرح شده در RFP را مشخص می‌کند. ستون «نحوه پوشش» چگونگی پوشش نیازمندی را مشخص می‌کند که اشاره به یک بخش از این سند و یا یکی از قابلیت‌های سامانه پیشنهادی دارد که با کد آن مشخص شده است.

جدول (۸-۱) نحوه پوشش نیازمندی‌های RFP توسط پیشنهاد پروژه

ردیف	صفحه RFP	موضوع بند RFP	نحوه پوشش
۱	۸	تامین SDK برای تعامل بین عامل و پلتفرم	قابلیت FT17
۲	۸	اعلام مشخصات سخت‌افزار مورد نیاز	جدول (۳-۱)
۳	۸	عدم وابستگی به پروتکل لایه MAC	قابلیت FT01 قابلیت FT02
۴	۸	تامین API برای برنامه‌های کاربردی	قابلیت FA02 قابلیت FA03 قابلیت FA04 قابلیت FA05
۵	۹	تعیین پروتکل ارتباط با تحلیل داده	قابلیت FA10 قابلیت FA12
۶	۹	پشتیبانی ۱۸ ماه	در نظر گرفتن این نیرو در تحلیل هزینه‌های پروژه
۷	۹	تخصیص تیم پیشنهاد دهنده	جدول (۴-۱)
۸	۹	تعیین متدولوژی توسعه نرم‌افزار	بخش ۳-۶-۱- بخش ۳-۶-۲-
۹	۱۰	نیازسنجی بر اساس RFP و پلتفرم‌های رایج	بند ۲-۲ شرح خدمات
۱۰	۱۰	طراحی بر اساس اصول مهندسی نرم‌افزار	بخش ۳-۶-۱- بخش ۳-۶-۲- بخش ۳-۶-۳-
۱۱	۱۰	تامین API برای برنامه‌های کاربردی	قابلیت FA02 قابلیت FA03 قابلیت FA04 قابلیت FA05
۱۲	۱۰	تعیین فرآیند تست	بخش ۳-۶-۳- بند ۴ شرح خدمات
۱۳	۱۰	ارایه الگوی معماری	بخش ۳-۵-۱-

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۴۶

ردیف	صفحه RFP	موضوع بند RFP	نحوه پوشش
۱۴	۱۰	امکان برنامه نویسی در سمت کاربر	قابلیت FA02 قابلیت FA03 قابلیت FA04 قابلیت FA05
۱۵	۱۱	پشتیبانی از پروتکل های LoRaWAN و NB-IoT و بومی	قابلیت FT01 قابلیت FT02 قابلیت FT03
۱۶	۱۱	قابلیت پیاده سازی پروتکل اختصاصی	قابلیت FT02 قابلیت FT04 قابلیت FT05
۱۷	۱۱	امکان get و set و گزارش دوره ای	قابلیت FT06 قابلیت FT07 قابلیت FT08
۱۸	۱۱	تامین SDK	قابلیت FT17
۱۹	۱۱	ثبت دستی اشیاء	قابلیت FT10
۲۰	۱۱	ثبت خودکار اشیاء	قابلیت FT10
۲۱	۱۱	مدیریت اشیا	قابلیت FT11 قابلیت FT12 قابلیت FT13 قابلیت FT14 قابلیت FT15
۲۲	۱۱	دریافت و ذخیره داده از طرف اشیاء	قابلیت FT16 قابلیت FA10
۲۳	۱۱	نمایش گرافیکی داده	قابلیت FA11
۲۴	۱۲	امکان اتصال به ابزارهای تحلیل داده	قابلیت FA12
۲۵	۱۲	پیاده سازی پروتکل استاندارد ارتباط با کاربردها	قابلیت FA01
۲۶	۱۲	امکان پیاده سازی پروتکل اختصاصی ارتباط با کاربردها	قابلیت FA06
۲۷	۱۲	امکان ارسال رخداد	قابلیت FA05
۲۸	۱۲	امکان تعریف مدل اطلاعاتی برای اشیاء	قابلیت FA07 قابلیت FA09
۲۹	۱۲	اعمال مدل اطلاعاتی بر روی تبادلات با اشیاء	قابلیت FA08
۳۰	۱۲	قابلیت پرداخت و کنترل دسترسی	قابلیت FA14

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۴۷

ردیف	صفحه RFP	موضوع بند RFP	نحوه پوشش
			قابلیت FA15 قابلیت FA16
۳۱	۱۲	قابلیت توسعه پذیری	قابلیت NF01 قابلیت NF02 قابلیت NF03 بخش ۳-۵-۲-
۳۲	۱۲	تمهیدات و پروتکل های امنیتی	قابلیت NF05 قابلیت NF06 قابلیت NF07 قابلیت NF08 قابلیت NF09 قابلیت NF10
۳۳	۱۲	تحمل پذیری خرابی	قابلیت NF01 قابلیت NF04 قابلیت NF11
۳۴	۱۳	مستقل از کاربرد	قابلیت FT02 قابلیت FT03 قابلیت FT05 قابلیت FT17 قابلیت FA01 قابلیت FA02 قابلیت FA06 قابلیت FA07 قابلیت FA08 قابلیت FA09
۳۵	۱۳	توانایی پشتیبانی از ۵۰۰۰ نود	قابلیت NF01 قابلیت NF02 قابلیت NF03
۳۶	۱۳	طراحی ماژولار	قابلیت NF04
۳۷	۱۳	مقیاس پذیری	قابلیت NF01 قابلیت NF02 قابلیت NF03 بخش ۳-۵-۲-

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۴۸

ردیف	صفحه RFP	موضوع بند RFP	نحوه پوشش
۳۸	۱۳	امنیت ارتباطات	قابلیت NF05 قابلیت NF06
۳۹	۱۳	طرح امنیت	قابلیت NF05 قابلیت NF06 قابلیت NF07 قابلیت NF08 قابلیت NF09 قابلیت NF10 بند ۲-۵-۱ شرح خدمات
۴۰	۱۴	طرح قابلیت اطمینان	قابلیت NF11 بند ۳-۵-۲ شرح خدمات
۴۲	۱۴	تخمین سخت افزار لازم	جدول (۱-۳)
۴۳	۱۴	نیازمندی های فرای RFP	قابلیت FT03 قابلیت FT04 قابلیت FA09 قابلیت FA10 قابلیت FA14 قابلیت NF08 قابلیت NF09

کد سند	تاریخ	صفحه
IoT-PF-PM-0010-v1.00	۹۶/۵/۸	۴۹