|  |  |  |
| --- | --- | --- |
|  | دانشکده مهندسی کامپیوتر و فناوری اطلاعات | F:\Arm\AKUT.svg.png |

آزمایشگاه اینترنت اشیاء

گروه پلتفرم

**عنوان سند:**

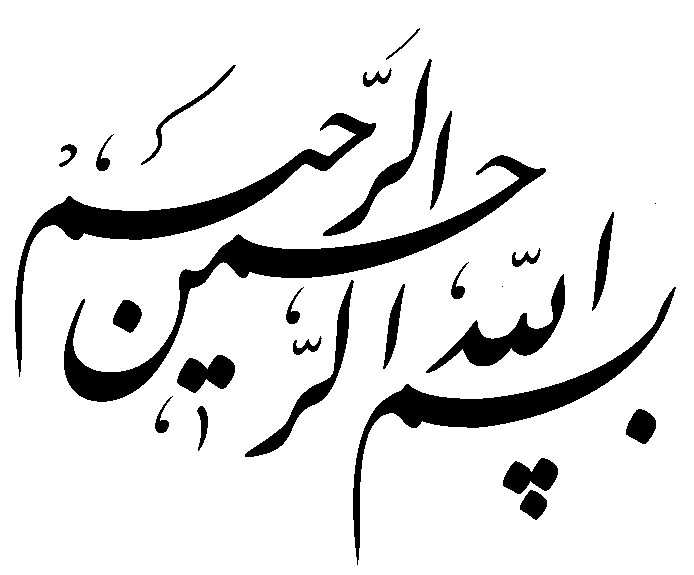
**تحليل نيازمندی‌های کارکردی پلتفرم اينترنت اشياء (تحلیل داده و حسابرسی)**

**كد سند:**

**IoT-RA-DP-v1.0**

**تاريخ:**

**15/01/97**



اطلاعات سند

|  |  |
| --- | --- |
| نام پروژه: | طراحي و پياده‌سازي پلتفرم اينترنت اشياء |
| نام سند: | تحليل نيازمندی‌های کارکردی پلتفرم اينترنت اشياء (تحلیل داده و حسابرسی) |
| کد سند: | IoT-RA-DP-v1.0 |
| وضعيت: | نهایی |
| تاريخ انتشار نهايي: | --- |
| نوع طبقه‌بندي سند: | محرمانه |

تاریخچه بازبینی‌ها

|  |  |  |  |
| --- | --- | --- | --- |
| ردیف | نام بازبینی کننده | تاریخ بازبینی | تغييرات |
| 1 | تيم فني | 12/8/96 | تهيه نسخه اوليه بر اساس توافقات جلسه 8/8/96 |
| 2 | تیم فنی | 15/01/97 | ویرایش فنی و ادبی |
| 3 |  |  |  |
| 4 |  |  |  |

چکيده

این سند در راستای پوشش فاز 2 پروژه پلتفرم اینترنت اشیا تهیه شده است. در این فاز تحلیل نیازمندی­های کارکردی و غیر کارکردی پلتفرم مد نظر می­باشد که با توجه به زمان­بندی ارائه شده، این سند نیازمندی‌های کارکردی پلتفرم اینترنت اشیاء در بخش پرداخت و تحلیل داده را شامل می­شود. در بخش تحلیل داده، نیازمندی­های مرتبط به اتصال به سیستم­های آنالیز داده شرح داده شده است. در سیستم پرداخت نیز نحوه ارتباطات مالی کاربر با پلتفرم و مدیریت پرداخت بررسی شده است.

فهرست مطالب

[فصل 1: نيازمندي‌هاي حسابرسی و تحلیل داده 7](#_Toc514699132)

[1-1- سرویس اتصال به ابزار تحلیل داده 7](#_Toc514699133)

[1-1-2 مزایای Apache Spark و نیازمندی­ها 11](#_Toc514699134)

[2-1 سرویس حسابرسی 12](#_Toc514699135)

فهرست اشکال

[شکل (1-1) نمای کلی ارتباطات در سرویس تحلیل داده 8](#_Toc498425423)

فهرست جداول

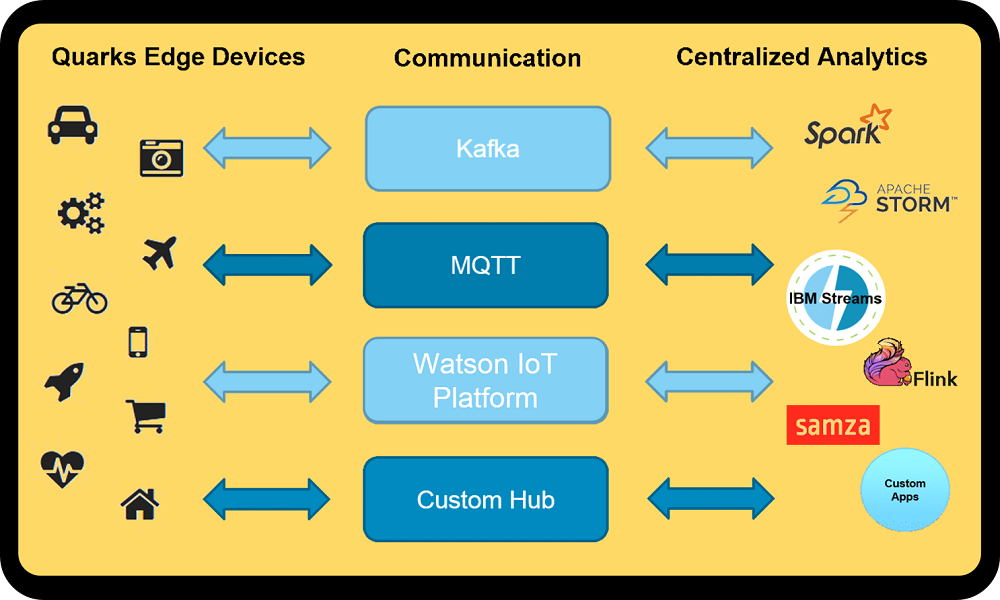
[جدول (1-1) نيازمندي‌هاي سرویس ارتباط با تحلیل داده 12](#_Toc498425431)

[جدول (1-2) نيازمندي‌هاي سرويس پرداخت 13](#_Toc498425432)

1. نيازمندي‌هاي حسابرسی و تحلیل داده
   1. سرویس اتصال به ابزار تحلیل داده

با شروع به کار پلفترم اینترنت اشیا حجم عظیمی از داده­ها از سمت اشیا ارسال خواهد گردید. در راستای برآورده کردن نیاز به پردزاش حجم عظیم داده­ها، سامانه باید امکان اتصال به ابزارهای تحلیل داده را فراهم کند. نیازمندی­های سرویس ارتباط با ابزارهای تحلیل داده در این بخش شرح داده شده است. این سرویس در واقع محل اتصال پلتفرم اینترنت اشیاء به ابزارهای تحلیل داده است که داده­های جمع­آوری شده از سنسورها را از منابع داده­ای مانند پایگاه داده گرفته و جهت تحلیل در اختیار این ابزارها قرار می­دهد.

جایگاه سرویس اتصال به ابزار تحلیل داده در ‏شکل (1-1) با عنوان Communication نشان داده شده است. این سرویس، مسئول ارتباط با ابزارهای تحلیل داده است که داده های جمع‌آوری شده از Deviceها را گرفته و به ابزارهای تحلیل داده تحویل می‌دهد. البته در طرح فعلی، داده­ها از منابع داده­ای پلتفرم در اختیار ابزار تحلیل داده قرار می­گیرد.



نمای کلی ارتباطات در سرویس تحلیل داده

تکنیک­های مختلفی برای تحلیل داده در اینترنت اشیا تعریف شده است که دارای دسته بندی­های مختلفی می­باشند. یک دسته بندی کلی به شرح زیر می­باشد.

* **تکنیک­های تحلیل اینترنت اشیا**

1. **تحلیل بلادرنگ** : معمولا روی داده‌های جمع‌آوری شده از سنسورها انجام می‌گیرد. در این حالت داده به طور مداوم تغییر می‌کند و تکنیک‌های تحلیل سریع داده برای خروجی تحلیلی در زمان کوتاه مورد نیاز است. دو معماری خوشه‌های پردازش موازی با استفاده از پایگاه‌های داده سنتی و پلتفرم‌های محاسباتی بر مبنای حافظه برای تحلیل بلادرنگ ارائه شده‌اند. Greenplum و Hana مثال‌هایی از معماری تحلیل بلادرنگ می‌باشند.
2. **تحلیل آفلاین** : زمانی استفاده می‌شود که یک پاسخ سریع نیاز نباشد. به عنوان مثال بسیاری از سازمان‌های اینترنت از معماری تحلیل آفلاین بر اساس هدوپ هزینه تبدیل فرمت داده را کاهش می‌دهند. SCRIBE، Kafka ، Time-Tunnel و Chukwa مثال‌هایی از معماری تحلیل آفلاین هستند و می‌توانند نیازهای کسب داده را برآورده کنند.
3. **تحلیل کلان** : زمانی که اندازه داده بزرگ‌‌تر از ظرفیت محصول تحلیل داده هوش تجاری و پایگاه‌‌های داده سنتی باشد از این تکنیک استفاده می­شود. تحلیل کلان از سیستم فایل توزیع‌‌شده هدوپ برای ذخیره داده و نگاشت/کاهش برای تحلیل داده استفاده می‌‌کند. علاوه بر این تحلیل کلان داده‌‌های دقیق را گرفته و ریسک‌‌های تصمیم‌‌های تجاری را کاهش می‌‌دهد. ابزارهای تحلیل مانند MapReduce، ‌‌ Spark ، Splunk و Skytree می‌‌توانند مجموعه داده‌‌های کلان اینترنت اشیاء را تحلیل کنند.

از میان ابزارهای تحلیل بلادرنگ موجود ابزار Apache Storm، از میان ابزارهای تحلیل آفلاین موجود ابزار Apache Chukwa و از میان ابزارهای تحلیل کلان نیز Apache Spark (همراه کتابخانه MLlib) به دلیل کاربری بالا، مقبولیت و متن باز بودن بررسی شده است.

* **ابزار Apache Storm**

Apache Stormیک سیستم محاسبات بلادرنگ توزیع­شده متن­باز و رایگان است. این سیستم ساده است و می­تواند با هر زبان برنامه­نویسی استفاده شود. موارد کاربردی از جمله تحلیل بلادرنگ، یادگیری ماشین آنلاین و محاسبات مستمر دارد. Storm سریع است و می­تواند یک ملیون چندتایی را در هر گره در ثانیه انجام دهد. این ابزار مقیاس­پذیر، متحمل خطا، متضمن پردازش داده و آسان در راه­اندازی و بکارگیری است. یک توپولوژی Storm، جریان­های داده را گرفته و پردازش می­کند.

Storm با هر سیستم صف­بندی و هر پایگاه­داده ای یکپارچه می­شود. به عنوان مثال می­توان آن را با Kestrel، RabbitMQ / AMQP، Kafka، JMS و Amazon Kinesis یکپارچه کرد. همچنین اتصال آن به پایگاه­داده آسان است و تنها باید یک ارتباط با پایگاه­داده مثل معمول برقرار کرد. Storm موازی­سازی، قسمت­بندی و تلاش مجدد در خطا را خود به کار می­برد.

* **ابزار Apache Chukwa**

یک سیستم جمع‌‌آوری داده متن‌‌باز برای نظارت بر سیستم‌‌های توزیعی بزرگ است که روی سیستم فایل توزیعی هدوپ (Hadoop Distributed File System) و فریم‌‌ورک نگاشت/کاهش ساخته شده و از مقیاس‌‌پذیری و مقاومت هدوپ ارث می­برد. این ابزار شامل مجموعه ابزار قدرتمند و انعطاف‌‌پذیر برای نمایش، نظارت و تحلیل نتایج برای استفاده از داده‌‌های جمع‌‌آوری شده است.

اهداف Chukwa:

* جمع‌‌آوری فایل‌‌های Log
* تاخیر در حدود دقیقه و کمتر از چند ساعت در جمع‌‌آوری داده
* مقیاس‌‌پذیری برای جمع‌‌آوری داده
* ذخیره‌‌سازی حجم عظیم داده در یک محل
* تحلیل پیشرفته log و داده‌‌کاوی
* فریم‌‌ورک گزارش‌‌دهی

Chukwa باید روی هر پلتفرم POSIX کار کند اما Linux تنها محصولی است که روی آن تست شده همچنین به طور موفقیت­آمیز روی Mac OS X استفاده شده است. تنها نیازمندی Java ورژن 1.6 به بعد و هدوپ ورژن 0.20.205.0 به بعد است. HICC رابط نمایش Chukwa است که به HBase ورژن 0.90.4 نیاز دارد.

* **ابزار Apache Spark**

Spark یک سیستم محاسبات خوشه عام‌‌ منظوره و سریع است که واسط‌‌های برنامه‌‌نویسی به زبان Java، Scala، Python و R و یک موتور بهینه پشتیبان گراف‌‌های عمومی اجرا را ارائه می‌‌دهد. Spark از یک مجموعه ابزار سطح بالاتر شامل Spark SQL برای SQL و پردازش داده ساخت‌‌یافته، MLlib برای یادگیری ماشین، Graphx برای پردازش گراف و Spark Streaming پشتیبانی می‌‌کند.

Spark روی ویندوز و سیستم‌‌های بر مبنای UNIX (مانند Linux و Mac OS) اجرا می‌‌شود. اجرای محلی روی یک سیستم آسان است و تنها باید Java روی سیستم نصب باشد. Spark از جاوا نسخه 8 به بعد، Python نسخه 2.7 و 3.4 به بعد و R نسخه 3.1 به بعد پشتیبانی می‌‌کند. Spark نسخه 2.2.0 از Scala 2.11 استفاده می‌‌کند.کتابخانه MLlib کتابخانه یادگیری ماشین Spark است که هدف آن تسهیل و مقیاس‌پذیر کردن یادگیری ماشین به صورت عملی است. این ابزار در سطح بالا موارد زیر را ارائه می‌دهد:

* الگوریتم‌های یادگیری ماشین: الگوریتم‌های معمول یادگیری ماشین مانند طبقه‌بندی، رگرسیون، خوشه‌بندی و فیلتر مشارکتی.
* ویژگی سازی : استخراج ویژگی، تبدیل، کاهش ابعاد و انتخاب ویژگی
* خطوط لوله : ابزارهایی برای ساخت، ارزیابی و تنظیم خطوط لوله یادگیری ماشین
* خدمات : جبر خطی، آمار، داده گردان و غیره….
  + 1. مزایای Apache Spark و نیازمندی­ها

Apache Spark یک موتور سریع و عمومی برای پردازش داده با مقیاس بزرگ است. با توجه به ویژگی­های مثبتی که این ابزار دارد به عنوان سرویس بخش ارتباطات تحلیل داده انتخاب شده است. از ویژگی‌های مثبت آن می‌توان به موارد زیر اشاره کرد:

* سرعت: برنامه‌ها را نسبت به Hadoop MapReduce تا 100 برابر سریع‌تر در حافظه و تا 10 برابر سریعتر بر روی دیسک اجرا می­کند. دارای یک موتور پردازش DAG است که از جریان داده غیرمدور و محاسبات درون حافظه‌ای پشتیبانی می‌کند.
* راحتی در استفاده: برنامه‌ها را می‌توان به زبان Java، Scala، Python و یا R نوشت. بیش از 80 عملیات سطح بالا ارائه می‌کند که ساخت برنامه‌های موازی را آسان می‌کند و می‌توان از آن به صورت تعاملی در پوسته‌های Scala، Python و R استفاده کرد.
* عمومیت: دستورات SQL، جریان سازی و تحلیل پیچیده را ترکیب می‌کند. Spark از یک پشته­ی کتابخانه‌ها شامل SQL و MLlib برای یادگیری ماشین، GraphX و Spark Streaming بهره می‌برد که می‌توانند در یک برنامه ترکیب شوند.
* اجرا در همه‌جا: Spark روی Hadoop، Mesos، به طور مستقل یا در ابر اجرا می‌شود و می‌تواند به منابع‌ داده شامل HDFS، Cassandra، HBase و S3 دسترسی پیدا کند. می‌توان Spark را با استفاده در حالت خوشه مستقل بر روی EC2، Hadoop YARN و Apache Mesos اجرا کرد. همچنین می‌توان به داده­ها در HDFS، Cassandra، HBase، Hive، Tachyon و یا هر منبع داده هدوپ دسترسی یافت.

نیازمندی­های مربوط به قسمت سرویس اتصال به ابزارهای تحلیل داده سامانه ‏جدول (1-1) آورده شده است:

نيازمندي‌هاي سرویس ارتباط با تحلیل داده

| **ردیف** | **کد** | **نیازمندی** | **ملاحظات** |
| --- | --- | --- | --- |
| ۱ | DTA-1 | ارتباط Spark (ابزار انتخاب شده) به پلتفرم فراهم گردد. | API و تنظیمات لازم در پلتفرم انجام می‌شود. |
| 2 | DTA-2 | داده ها به صورت سریع در آن ذخیره و بازیابی کرد. | استفاده از پایگاه داده NoSQL |
| 3 | DTA-3 | خواندن و ارسال انواع داده (Structured و Unstructured) پشتیبانی گردد. | با ابزار Spark بر آورده می‌شود. ارتباط با پایگاه ‌داده­های NoSQL و SQL |
| 4 | DTA-4 | برای پوشش سایر ابزارهای تحلیل از زبان جاوا ورژن 8 به بعد استفاده شود. | --- |
| ۵ | DTA-5 | برای داده های حجیم مقیاس پذیر باشد. | با ابزار Spark بر آورده می‌شود |

* 1. سرویس حسابرسی

با توجه به اینکه درآمدزایی پلتفرم اینترنت اشیا از طریق فروش سرویس به کاربران (فروش اشیا) می­باشد پلتفرم مورد نظر باید قابلیت انجام پرداخت از سمت کاربران و مدیریت پرداخت­ها و تراکنش­های کاربران را داشته باشد. موارد قابل ذکر در تحلیل نیازمندی­های سرویس پرداخت به شرح زیر است:

* نحوه محاسبه هزینه برای مشتریان

با توجه به بررسی­های انجام شده، پلتفرم­های مختلف سیاست­های مختلفی در این زمینه در پیش می­گیرند که وابسته به مدل تجاری[[1]](#footnote-1) آن­ها است. عمده روش­های اعمال هزینه به شرح زیر می­باشد:

1. به ازای هر سنسور
2. به ازای دیتای رد و بدل شده
3. به ازای زمان
4. ترکیبی از روش­های فوق

با توجه به بررسی انجام شده و همچنین دریافت نظرات کارفرما ترکیبی از تعداد سنسور در مدت زمان استفاده جز گزینه­های نهایی مطرح گردید که با استفاده از پکیچ­هایی که در آن­ها تعداد سنسور و زمان استفاده از آن متغیر هستند تعریف می­گردند. این پکیج­ها توسط مدیر پلتفرم قابل تعریف می­باشد. همچنین کاربران می­توانند اکانت رایگان یک یا دو ماهه برای یک یا دو سنسور نیز داشته باشند که در هنگام ورود به صورت پیش­فرض برای آن­ها فعال می­گردد.

* نوع پرداخت :

در حالت کلی کاربران بر اساس دو نوع postpaid و prepaid می­توانند پرداخت را انجام دهند. با توجه به اینکه پکیج انتخابی به صورت تعداد سنسور در ماه می­باشد و جهت اعمال کنترل­های لازم نوع پرداخت prepaid می­باشد.

* نحوه ارتباط با سیستم­های بانکی

عموما دو نوع وب سرویسی (مانند آپ) و پرداخت از طریق درگاه بانکی در کشور استفاده می­شود. پرداخت از طریق درگاه بانکی با توجه به محددیت­های زمانی پروژه بهترین و ایمن­ترین گزینه است.

نیازمندی­های اصلی و نهایی سرویس پرداخت در ‏جدول (1-2) شرح داده شده است:

نيازمندي‌هاي سرويس پرداخت

| **ردیف** | **کد** | **نیازمندی** | **ملاحظات** |
| --- | --- | --- | --- |
| 1 | Pay-1 | نحوه پرداخت هزینه مشترکین از طریق درگاه بانکی باشد. |  |
| 2 | Pay-2 | نحوه محاسبه هزینه مشترکین به صورت پکیج­های تعداد سنسور در زمان استفاده باشد. |  |
| 3 | Pay-3 | نوع پرداخت به صورت prepaid باشد. |  |
| 4 | Pay-4 | نماد الکترونیک برای سایت جهت راه اندازی درگاه پرداخت از سمت پژوهشگاه فراهم گردد. |  |
| 5 | Pay-5 | دو نوع کاربر Permium و free وجود داشته باشد. امکان استفاده کاربران به صورت رایگان برای بازه یک یا دو ماهه برای یک یا دو سنسور وجود داشته باشد. |  |
| 6 | Pay-6 | کاربر توانایی تولید گزارشات از بسته­های خریداری شده و تراکنش­ها را دارد. | سایر جزییات در IoT-RA-UI-v1.0 ذکر شده است |
| 7 | Pay-7 | ادمین پلتفرم توانایی مدیریت سیستم پرداخت شامل تولید گزاراشات از اطلاعات جمع آوری شده و مدیریت بسته­ها را داشته باشد. | سایر جزییات در IoT-RA-UIM-v1.0 ذکر شده است |

1. Businnes Model [↑](#footnote-ref-1)