



پژوهشگاه فضایی ایران

مرکز تحقیقات فضایی

عنوان اختصاری پروژه: اینترنت اشیاء کشاورزی

کد پروژه: CNS۹۹۹۷-۰۱

کد فعالیت: CNS۹۹۹۷-۰۱-۰۷

کتابخانه‌های ارائه شده برای اسکریت نویسی

کد سند: MSRI-CNS۹۹۹۷-۰۱/R-۰۷/۱۱

شماره ویرایش: ۱/۱

طبقه‌بندی: عادی

تاریخ: ۹۷/۰۴/۱۶

تعداد کل صفحات: ۱۸ صفحه

(با احتساب برگ روی جلد)

استفاده از این سند صرفاً توسط گیرندگان مجاز است.

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

کد سند: MSRI-CNS۹۹۹۷-۰۱/R-۰۷/۱۱ ویرایش: ۱/۱	کتابخانه‌های ارائه شده برای اسکرپت نویسی	 پژوهشگاه فضایی ایران
طبقه‌بندی: عادی		

شناسنامه سند

۱- مشخصات پروژه

عنوان کامل پروژه	انجام مطالعات امکانسنجی و پیاده سازی ایده اینترنت اشیا در حوزه کشاورزی			مدیر پروژه	-
کد پروژه	CNS۹۹۹۷-۰۱	پژوهشکده (حوزه) مجری	مرکز تحقیقات فضایی	تاریخ شروع پروژه	۹۵/۰۲/۰۱
کد فعالیت	CNS۹۹۹۷-۰۱-۰۷			تاریخ خاتمه پروژه	۹۵/۰۹/۰۱

۲- مشخصات سند

عنوان سند	تحلیل کتابخانه‌های ارائه شده برای اسکرپت نویسی			تعداد صفحات	
طبقه‌بندی سند	عادی	کد سند	MSRI-CNS۹۹۹۷-۰۱/R-۰۷/۱۱	کل سند	۱۸
ویرایش	۱/۱	تاریخ ویرایش	۹۷/۰۴/۱۶	پیوست‌ها	۱۱

۳- جدول تهیه، تایید و تصویب در پژوهشکده (حوزه) مجری

سمت*	نام و نام خانوادگی	امضا	تاریخ
تهیه‌کننده (گان)	رئیس بخش طراحی و پیاده سازی نرم افزار		
تاییدکننده (گان)	رئیس اداره برنامه ریزی		
تصویب‌کننده	رئیس مرکز		

* برای مواردی که مجری، حوزه دیگری غیر از پژوهشکده است، مثل مراکز یا گروه‌های پژوهشی مستقل و ... از سمت‌های معادل بر اساس نظر رئیس حوزه استفاده شود.

شناسنامه سند (ادامه)

۴- جدول تایید و تصویب در پژوهشگاه

تاریخ	امضا	نام و نام خانوادگی	سمت	
			مدیران مرکز طراحی و توسعه سامانه‌های فضایی یا مدیر پژوهش و فناوری (برحسب مورد)	تاییدکننده (گان)
			سایر افراد (مانند معاون تضمین کیفیت، بهره‌بردار و ... براساس قرارداد یا نظر تصویب‌کننده و مدیریت کنترل پروژه)	
			رییس مرکز طراحی و توسعه سامانه‌های فضایی یا معاون پژوهش و فناوری (برحسب مورد)	تصویب‌کننده

۵- جدول توزیع نسخ (گیرندگان)

توزیع	عنوان واحد	توزیع	عنوان واحد
	مدیریت راهبرد و طراحی مأموریت		ریاست پژوهشگاه فضایی ایران
	مدیریت مهندسی سامانه‌های فضایی		دفتر ریاست، روابط عمومی و امور بین الملل
	مدیریت آزمون و عملیات میدان		مدیریت حراست
	معاونت اجرایی		اداره امور حقوقی
	مدیریت توسعه منابع انسانی		مدیریت نظارت و ارزیابی و پاسخگویی به شکایات
	مدیریت پشتیبانی		مدیریت بازرگانی خارجی
	مدیریت امور مالی	●	مدیریت طرح و برنامه
	اداره تشکیلات و بهبود روش‌ها		معاونت پژوهش و فناوری
	سازمان فضایی ایران		مدیریت آموزش و تحصیلات تکمیلی
	پژوهشکده سامانه‌های حمل و نقل فضایی	✓	مدیریت پژوهش و فناوری
	پژوهشکده سامانه‌های ماهواره		مدیریت دانش
	پژوهشکده مکانیک		معاونت تضمین کیفیت و ایمنی
	پژوهشکده مواد و انرژی		مدیریت مهندسی تضمین کیفیت
	پژوهشکده رانشگرهای فضایی		مدیریت کالیبراسیون و استاندارد
✓	مرکز تحقیقات فضایی		مدیریت ایمنی و محیط زیست
			مرکز طراحی و توسعه سامانه‌های فضایی
			سایر گیرندگان:

*توزیع نسخ بر اساس علامت‌های زیر انجام می‌شود:

✓: سند برای این واحدها ارسال می‌شود.

●: سند برای این واحدها ارسال نمی‌شود و صرفاً اطلاع‌رسانی می‌شود.

۶- تایید مرکز اسناد

مدیریت دانش (مرکز اسناد) پژوهشگاه فضایی ایران
نام و نام خانوادگی:
تاریخ:
مهر و امضا

مدیریت دانش (مرکز اسناد) پژوهشکده مجری
نام و نام خانوادگی:
تاریخ:
مهر و امضا

شناسنامه سند (ادامه)

۷- جدول مشخصات و شرح وظایف دست‌اندرکاران تدوین سند*

ردیف	نام و نام خانوادگی	آخرین مدرک تحصیلی	رشته تحصیلی	مرتبه علمی**	محل کار	شرح وظایف	درصد مشارکت
۱	دانشگاه امیرکبیر				دانشگاه امیرکبیر		۱۰۰
۲							
۳							
۴							
۵							
۶							
۷							
۸							
۹							
جمع							۱۰۰

*منظور کلیه افرادی است که در انجام فعالیت‌های مرتبط با این سند نقش اصلی داشته‌اند.

**برای اعضای هیات علمی از عناوین مربوط (استاد، دانشیار، استادیار، مربی) و برای دیگر پژوهشگران از عنوان کارشناس استفاده شود.

۸- دیگر همکاران تدوین سند*

ردیف	نام و نام خانوادگی	آخرین مدرک تحصیلی	رشته تحصیلی	مرتبه علمی	محل کار	نقش

* منظور کسانی است که ضمن مطالعه سند، نظرات قابل توجهی را در خصوص سند ارائه کرده‌اند. ویراستاران ادبی نیز در این جدول ذکر می‌شوند.

کد سند: MSRI-CNS۹۹۹۷-۰۱/R-۰۷/۱۱ ویرایش: ۱/۱ طبقه بندی: عادی	کتابخانه های ارائه شده برای اسکریت نویسی	 پژوهشگاه فضایی ایران
-------------------------------------------------------------------	------------------------------------------	------------------------------------------------------------------------------------------------------------

شناسنامه سند (ادامه)

۹- جدول مشخصات ناظر(ان)

ردیف	نام و نام خانوادگی	آخرین مدرک	رشته	مرتبه علمی	محل کار	توضیحات
۱	احسان پناهی	فوق لیسانس		کارشناس	مرکز تحقیقات فضایی	

۱۰- جدول سوابق ویرایش و تغییرات

ویرایش	تاریخ	شرح تغییرات	علت / مرجع تغییر	واحد تهیه کننده مسئول
۱/۱	۹۶/۰۴/۱۶	نگارش سند	-	-

کد سند: MSRI-CNS۹۹۹۷-۰۱/R-۰۷/۱۱ ویرایش: ۱/۱	کتابخانه‌های ارائه شده برای اسکرپت نویسی	 پژوهشگاه فضایی ایران
طبقه‌بندی: عادی		

چکیده

گزارش حاضر حاوی سند دریافتی از دانشگاه صنعتی امیرکبیر (پیمانکار پژوهشگاه فضایی ایران در قرارداد "طراحی و پیاده سازی پلتفرم اینترنت اشیا") با عنوان "کتابخانه‌های ارائه شده برای اسکرپت نویسی" می باشد.

واژه‌های کلیدی:

اینترنت اشیا، پلتفرم، کشاورزی هوشمند



دانشکده مهندسی کامپیوتر و فناوری اطلاعات



آزمایشگاه اینترنت اشیا

گروه پلتفرم

عنوان سند:

کتابخانه‌های ارائه شده برای اسکریپت نویسی

کد سند:

IoT-Imp-UserLib-v1.1

تاریخ:

۹۷/۰۱/۱۴

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

حاشیہ
برائے
الشریفات

اطلاعات سند

طراحی و پیاده سازی پلتفرم اینترنت اشیا	نام پروژه:
کتابخانه های ارائه شده برای اسکریپت نویسی	نام سند:
IoT-Imp-UserLib-v۱.۱	کد سند:
نهایی	وضعیت:
---	تاریخ انتشار نهایی:
محرمانه	نوع طبقه بندی سند:

صفحه	تاریخ	کد سند
۱	۹۷/۰۱/۱۴	IoT-Imp-UserLib-v۱.۱

تاریخچه بازبینی‌ها

ردیف	نام بازبینی کننده	تاریخ بازبینی	تغییرات
۱	تیم فنی	۹۶/۹/۱۸	ویرایش بر اساس نظرات کارفرما
۲	تیم فنی	۹۶/۱۱/۲۰	افزودن کلاس های جدید
۳	تیم فنی	۹۷/۰۱/۱۴	ویرایش فنی و ادبی
۴			

صفحه	تاریخ	کد سند
ب	۹۷/۰۱/۱۴	IoT-Imp-UserLib-v۱.۱

چکیده

این سند جز اسناد فاز طراحی می‌باشد. در این سند طراحی کتابخانه‌هایی که در اسکریپت نویسی در اختیار کاربر قرار خواهد گرفت تعریف و تشریح شده‌اند. کاربر سیستم تنها توانایی استفاده از کتابخانه‌های تعریف شده در این سند را دارا خواهد بود.

صفحه	تاریخ	کد سند
ج	۹۷/۰۱/۱۴	IoT-Imp-UserLib-v۱.۱

فهرست مطالب

فصل ۱: کتابخانه‌های اسکریپت نویسی

۵

۱-۱- تعریف کتابخانه‌ها ۵

۱-۱-۱- کلاس Scenario ۵

۱-۱-۲- کلاس db_cr ۶

۱-۱-۳- کلاس notification_actions ۷

۱-۱-۴- کلاس ui_geopy_gps ۷۴۸

۱-۱-۵- سناریوهای تست کتابخانه ۷۴۸

صفحه	تاریخ	کد سند
د	۹۷/۰۱/۱۴	IoT-Imp-UserLib-v۱.۱

فصل ۱:

کتابخانه‌های اسکریپت نویسی

۱-۱- تعریف کتابخانه‌ها

با توجه به اینکه کاربران پلتفرم قابلیت اسکریپت نویسی دارند لیست کتابخانه‌هایی که یک کاربر می‌تواند در زبان پایتون استفاده کند در چهار کلاس Scenario، db_cr، notification_action و ui_geopy_gps تعریف شده‌اند که در ادامه جزئیات هر کلاس ارائه گردیده است. در نهایت سناریوهایی نیز برای تست این کتابخانه‌ها طراحی شده است که در انتهای گزارش ذکر شده‌اند.

۱-۱-۱- کلاس Scenario:

- کلاس Scenario شامل توابع دریافت داده، ارسال داده و توابع مربوط به زمان است.
- تابع sleep(seconds): اجرای کد را برای مدت زمان داده شده به تعویق می‌اندازد. واحد تاخیر آن بر حسب ثانیه است.
 - تابع schedule(delay_seconds, action_fuction, args): این تابع در ورودی میزان تاخیر (بر اساس ثانیه) ، تابع مورد نظر جهت اجرا و آرگومان‌های تابع را می‌گیرد. در ادامه پس از گذشت تاخیر ذکر شده در ورودی، تابع را اجرا می‌کند.

صفحه	تاریخ	کد سند
۵	۹۷/۰۱/۱۴	IoT-Imp-UserLib-v۱.۱

• تابع `wait_for_data(timeout)`:

این تابع برای دریافت داده است که می توان آن را به صورت سنکرون یا آسنکرون استفاده نمود. پارامتر آن به شرح زیر است:

۱. `timeout`: میزان زمانی که منتظر می ماند تا داده را دریافت کند (بر اساس ثانیه).

این تابع پاسخی که از سرور دریافت می کند را به عنوان خروجی برمی گرداند.

• تابع `send_to_down_link(message, timeout)`:

این تابع پیغام (داده) را روی یک پورت ارسال می کند و پیام تایید دریافت می کند. پارامترهای آن به شرح زیر است:

۱. `message`: محتوای پیامی که ارسال می کند.

۲. `timeout`: میزان زمانی که منتظر میماند تا داده را دریافت کند (بر اساس ثانیه).

پیام را ارسال می کند و پاسخ تاییدی را که از سرور دریافت می کند به عنوان خروجی برمی گرداند.

۱-۲-۱- کلاس `db_cr`:

این قسمت برای ارتباط با پایگاه داده در نظر گرفته شده است و بر اساس نظر کارفرما تنها عملیات خواندن و ایجاد مجاز شمرده می شوند.^۱

• تابع `create_one(document)`: یک سند به صورت دیکشنری پایتون گرفته و آن را در پایگاه داده ذخیره می کند و id آن را برمی گرداند.

• تابع `create_many(document_list)`: یک لیست از مستندات را گرفته و در پایگاه داده ذخیره کرده و لیست id آن ها را برمی گرداند.

• تابع `read_one(partial_document)`: قسمتی از یک سند را گرفته و اولین سند دارای آن قسمت را برمی گرداند.

• تابع `read_many(partial_document)`: قسمتی از یک سند را گرفته و همه اسناد دارای آن قسمت را برمی گرداند.

^۱ در فایل `db_config` تنظیمات مربوط به سرور پایگاه داده شامل `ip` و پورت سیستم اجرا کننده پایگاه داده و همچنین نام پایگاه داده موجود در آن (`db_name`) و `collection` شامل داده ها (`db_collection`) مشخص شده است که بعداً در اختیار مدیر پلتفرم قرار خواهد گرفت.

کد سند	تاریخ	صفحه
IoT-Imp-UserLib-v۱.۱	۹۷/۰۱/۱۴	۶

۱-۳-۱ notification_actions کلاس

این قسمت برای اطلاع رسانی رویدادهای سیستم به کاربر است.

- تابع `send_email(host, port, username, password, sender, receivers, message)`: به ترتیب ip هاست سرور ارسال ایمیل، پورت سرور ارسال ایمیل، نام کاربری، کلمه عبور، آدرس ایمیل ارسال کننده، لیست ایمیل دریافت کنندگان و متن ایمیل را گرفته و پیغام را از طریق سرور مشخص شده به لیست دریافت کنندگان ارسال می کند. در صورت موفقیت مقدار `True` برمیگرداند. در صورت عدم موفقیت و دریافت خطای `smtplib.SMTPException`، پیغام خطا را چاپ کرده و مقدار `False` برمی گرداند.
- تابع `send_sms(username, password, from_number, to_number, message)`: به ترتیب نام کاربری، کلمه عبور، شماره تماس ارسال کننده، شماره تماس دریافت کننده و متن پیام را گرفته و توسط پنل مشخص شده در کد، پیامک را ارسال می کند. در صورت موفقیت مقدار `True` برمیگرداند. در صورت عدم موفقیت پیام خطا چاپ می شود و مقدار `False` برمی گرداند.

۱-۴-۱ ui_geopy_gps کلاس

در این بخش از کتابخانه ی geopy به منظور دریافت اطلاعات جغرافیایی استفاده شده است.

- برای دریافت مختصات و آدرس کامل یک محل از `Nominatim().geocode("address")` استفاده میشود. خروجی آن شامل متغیر `address` به عنوان آدرس کامل، متغیر `latitude` عرض جغرافیایی، متغیر `longitude` طول جغرافیایی و متغیر `raw` به عنوان کل اطلاعات می باشد.
- برای دریافت آدرس با استفاده از مختصات از تابع `Nominatim().reverse(latitude, longitude)` استفاده می شود که متغیرهای `address`، `latitude`، `longitude` و `raw` مانند حالت قبل برگردانده می شوند.

۱-۵-۱ سناریوهای تست کتابخانه

در پوشه ی `scenarios`، سناریوهایی برای تست توابع تعریف شده در این سند طراحی و پیاده سازی شده اند. شایان ذکر است که این سناریوها تنها برای تست و آشنایی از نحوه کارکرد این توابع طراحی شده اند و کاربرد دیگری ندارند. سناریوهای تست به شرح زیر می باشند:

- سناریو یک- رخداد دریافت داده ی سنسور شماره "x" بر روی شی "a" از طریق ایمیل اطلاع رسانی گردد:

کد سند	تاریخ	صفحه
IoT-Imp-UserLib-v۱.۱	۹۷/۰۱/۱۴	۷

این سناریو از طریق تابع `wait_for_data` منتظر دریافت داده‌ی سنسور شماره `x` از شیء `a` می‌ماند. سپس در تابع `action` چک می‌کند که داده دریافت شده برای سنسور `x` از شیء `a` باشد که در صورت درست بودن، با استفاده از تابع `send_email` ایمیل را ارسال می‌کند.

- سناریو دو- اگر داده سنسور شماره `x` بر روی شیء `a` دریافت گردید، به اندازه `w` ثانیه منتظر بماند، آخرین مقدار سنسور `y` از شیء `b` را خوانده و با مقدار سنسور `x` جمع کرده و در پایگاه داده ذخیره کند: این سناریو با استفاده از تابع `wait_for_data` منتظر دریافت داده‌ی سنسور شماره `x` از شیء `a` می‌ماند. سپس در تابع `action` چک می‌کند که داده دریافت شده برای سنسور `x` از شیء `a` باشد و در این صورت `w` ثانیه منتظر می‌ماند. در ادامه با استفاده از تابع `read_one` آخرین مقدار سنسور `y` از شیء `b` را خوانده و با مقدار دریافت شده از سنسور `x` جمع می‌کند و در نهایت با استفاده از تابع `create_one` مقدار مجموع را در پایگاه داده ذخیره می‌کند.

- سناریو سه- اگر میانگین ۵ داده اخیر سنسور شماره `x` بر روی شیء `a` بزرگتر از `w` بود یک ایمیل بفرستد: تابع `read_data_from_db` با استفاده از تابع `read_many` ۵ داده اخیر سنسور `x` از شیء `a` را دریافت می‌کند. سپس میانگین این ۵ داده محاسبه می‌گردد و با مقدار `w` مقایسه می‌شود. در صورتی که مقدار میانگین از `w` بزرگتر باشد تابع `action` فراخوانی شده و در آن ایمیل اطلاع‌رسانی با استفاده از تابع `send_email` ارسال می‌گردد.

- سناریو چهار- اگر ترتیب دریافت داده به صورتی بود که در ابتدا داده سنسور `x` از شیء `a` آمد و سپس داده سنسور `y` از شیء `b` آمد، میانگین آنها را در پایگاه داده ذخیره کند: این سناریو با استفاده از تابع `wait_for_data` منتظر دریافت داده‌ی سنسور شماره `x` از شیء `a` می‌ماند. سپس در تابع `action` چک می‌گردد که داده دریافت شده برای سنسور `x` از شیء `a` باشد و در این صورت به حالت بعدی می‌رود. در این حالت نیز دوباره منتظر دریافت داده‌ی سنسور شماره `y` از شیء `b` می‌ماند. در ادامه در تابع `action` چک می‌کند که داده دریافت شده برای سنسور `y` از شیء `b` باشد و در این صورت به حالت نهایی می‌رود. در حالت نهایی میانگین داده‌های دریافتی را محاسبه کرده و با استفاده از تابع `create_one` داده‌ی جدید را در پایگاه داده ذخیره می‌کند.

- سناریو پنج- هر `w` ثانیه یک بار، میانگین ۵ داده اخیر سنسور `x` از شیء `a` را با میانگین ۵ داده اخیر سنسور `y` از شیء `a` مقایسه کند و اگر بزرگتر بود دستور `Z` را ارسال کند: هر بار با استفاده از تابع `read_many` در

کد سند	تاریخ	صفحه
IoT-Imp-UserLib-v۱.۱	۹۷/۰۱/۱۴	۸

تابع `read_from_db` ۵ داده اخیر سنسور X از شیء `a` و همچنین ۵ داده اخیر سنسور Y از شیء `a` را دریافت کرده و میانگین آن‌ها را محاسبه می‌کند. سپس آن‌ها را مقایسه کرده و اگر میانگین داده‌های سنسور X از سنسور Y بزرگتر بود، با استفاده از تابع `send_to_down_link` یک دستور ارسال می‌کند و سپس `w` ثانیه منتظر می‌ماند و دوباره از اول این عملیات تکرار می‌گردد.

صفحه	تاریخ	کد سند
۹	۹۷/۰۱/۱۴	IoT-Imp-UserLib-v۱.۱