



مرکز تحقیقات فضایی

عنوان اختصاری پروژه: شناسایی بیماری های گندم

کد پروژه: SAP9997-01

کد فعالیت: SAP9997-01-03

گزارش معماری سامانه پروژه شناسایی بیماری های گندم  
(فاز سوم پروژه)

کد سند: MSRI- SAP9997-01-R-03/01

شماره ویرایش: ۱/۲

طبقه بندی: عادی

تاریخ: ۱۳۹۷/۰۹/۲۷

تعداد کل صفحات: ۲۰ صفحه

(با احتساب برگ روی جلد)

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

## گزارش معماری سامانه پروژه شناسایی بیماری های گندم ( فاز سوم پروژه )

کد سند: MSRI- SAP9997-01-R-03/01  
ویرایش: 1/1  
طبقه بندی: عادی

### شناسنامه سند

#### ۱- مشخصات پروژه

عنوان کامل پروژه	طراحی و پیاده سازی سامانه شناسایی بیماری های گندم با استفاده از مدل سازی داده های سنسورهای زمینی و سنجش از دور	مدیر پروژه	داوود عاشورلو
کد پروژه	SAP9997-01	پژوهشکده (حوزه) مجری	مرکز تحقیقات فضایی
کد فعالیت	SAP9997-01-03	تاریخ شروع پروژه	۹۵/۱۲/۲۱
		تاریخ خاتمه پروژه	۹۶/۱۱/۲۱

#### ۲- مشخصات سند

عنوان سند	گزارش معماری سامانه پروژه شناسایی بیماری های گندم (فاز سوم پروژه)			تعداد صفحات
طبقه بندی سند	عادی	کد سند	MSRI- SAP9997-01-R-03/01	کل سند
ویرایش	1/1	تاریخ ویرایش	۱۳۹۷/۰۹/۲۷	پیوست ها
				۲۰

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

سمت*	نام و نام خانوادگی	امضا	تاریخ

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

## گزارش معماری سامانه پروژه شناسایی بیماری های گندم ( فاز سوم پروژه )

کد سند: MSRI- SAP9997-01-R-03/01  
ویرایش: 1/1  
طبقه بندی: عادی

### شناسنامه سند (ادامه)

#### ۴- جدول تایید و تصویب در پژوهشگاه

تاریخ	امضا	نام و نام خانوادگی	سمت	
			مدیران مرکز طراحی و توسعه سامانه های فضایی یا مدیر پژوهش و فناوری (برحسب مورد)	تایید کننده (گان)
			سایر افراد (مانند معاون تضمین کیفیت، بهره بردار و ... براساس قرارداد یا نظر تصویب کننده و مدیریت کنترل پروژه)	
			رئیس مرکز طراحی و توسعه سامانه های فضایی یا معاون پژوهش و فناوری (برحسب مورد)	تصویب کننده

#### ۵- جدول توزیع نسخ (گیرندگان)

توزیع *	عنوان واحد	توزیع *	عنوان واحد
	سازمان فضایی ایران		ریاست پژوهشگاه فضایی ایران
✓	پژوهشکده سامانه های حمل و نقل فضایی		معاونت پژوهش و فناوری
	پژوهشکده سامانه های فضانوردی		معاونت تضمین کیفیت و ایمنی
	پژوهشکده سامانه های ماهواره		معاونت اجرایی
●	پژوهشکده مکانیک		مدیریت طرح و برنامه
	پژوهشکده مواد و انرژی		مرکز طراحی و توسعه سامانه های فضایی
✓	پژوهشکده رانشگرهای فضایی		مرکز تحقیقات فضایی
	گروه پژوهشی حقوق فضا		
سایر گیرندگان:			

\* توزیع نسخ بر اساس علامت های زیر انجام می شود:

✓: سند برای این واحدها ارسال می شود.

●: سند برای این واحدها ارسال نمی شود و صرفا اطلاع رسانی می شود.

#### 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	پدرام شاه صفی	کارشناسی ارشد	فناوری اطلاعات	کارشناس	مرکز تحقیقات فضایی		۴۰
۲	سارا رجب زاده	کارشناسی ارشد	فناوری اطلاعات	کارشناس	مرکز تحقیقات فضایی		۴۰
۳	شاهرخ جلیلیان	کارشناسی ارشد	برق الکترونیک	مربی	مرکز تحقیقات فضایی		۲۰
۴							
۵							
۶							
۷							
۸							
۹							
۱۰۰	جمع						

\*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.

شناسنامه سند (ادامه)

۹- جدول مشخصات ناظر(ان)

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

۱۰- جدول سوابق ویرایش و تغییرات

ویرایش	تاریخ	شرح تغییرات	علت / مرجع تغییر	واحد تهیه‌کننده مسئول
۱/۱	۹۶/۱۱/۳۰	نگارش سند		
۱/۲	۹۷/۱۰/۱۶	افزودن کد		

## فهرست مطالب

۸	مقدمه
۸	مشخصات سامانه پیش‌بینی احتمال وقوع بیماری فوزاریوم گندم
۱۰	پایتون
۱۰	پایگاه داده
۱۱	جانگو
۱۱	پروتکل ارتباطی با سرورهای خارجی
۱۲	الگوریتم
۱۲	سرور ذخیره سازی تصاویر
۱۳	وب سایت هواشناسی
۱۳	سرورهای خارجی سرویس گیرنده
۱۶	رابط کاربری
۱۷	سیستم احراز هویت
۱۹	مدیر سیستم
۱۹	خروجی‌های پردازش شده
۲۰	ساختار کدها
۳۲	پیوست‌ها

## مقدمه

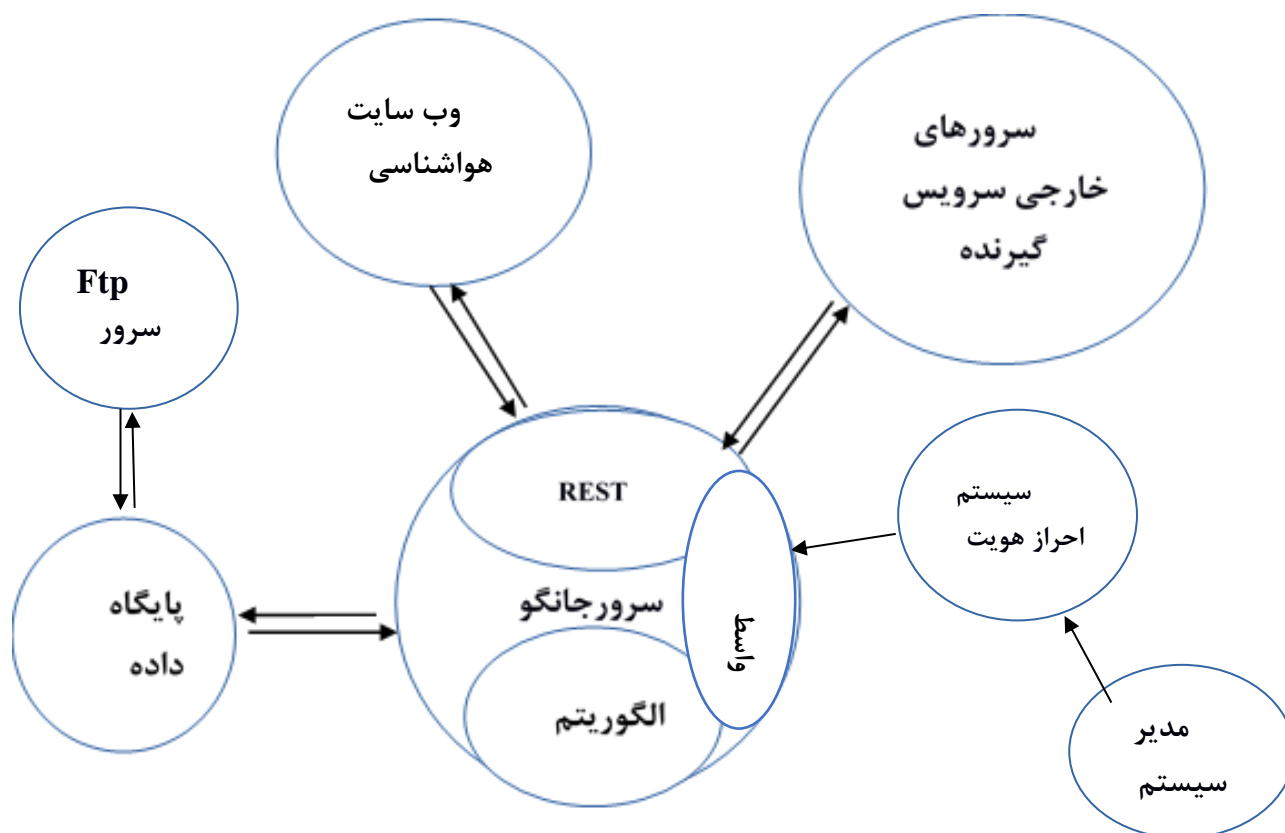
در این گزارش به ارائه اجمالی سامانه طراحی شده توسط تیم برنامه نویسی سنجش از دور در پروژه شناسایی بیماری فوزاریوم گندم پرداخته می‌شود. در ابتدا به معرفی ابزارهای مورد استفاده در پروژه خواهیم پرداخت، سپس معماری سامانه و مولفه‌های آن تشریح می‌شوند.

## مشخصات سامانه پیش‌بینی احتمال وقوع بیماری فوزاریوم گندم

سامانه بر اساس الگوریتم‌های پیاده‌سازی شده احتمال وقوع بیماری فوزاریوم گندم را پیش‌بینی می‌کند. برای پیاده سازی الگوریتم‌ها از زبان برنامه‌نویسی پایتون نسخه ۳٫۶ و فریم‌ورک جانگو استفاده شده است. پایگاه داده مورد استفاده در این پروژه PostgreSQL است.

معماری یکپارچه‌ای برای مدیریت و نگهداری داده‌ها فراهم شده که به صورت شکل زیر پیاده سازی شده است.





شکل ۱-۱ شمای یکپارچه از سیستم پیش‌بینی بیماری فواریوم گندم

اجزای این سامانه در زیربخش‌های زیر بطور مختصر شرح داده شده است:

## پایتون

پایتون یک زبان شفاف و قدرتمند شی‌گرا است که قابل مقایسه با زبان‌هایی مثل جاوا و پریل است. از جمله ویژگی‌های این زبان می‌توان به موارد زیر اشاره کرد:

- متن باز بودن.
- وجود منابع و کتابخانه‌های غنی.
- سرعت بالای توسعه.
- قابلیت اجرا بر روی پلتفرم‌ها و سیستم‌عامل‌های مختلف.

در پروژه پیش‌بینی بیماری فوزاریوم گندم تمام الگوریتم‌هایی که توسط تیم فنی سنجش از دور توسعه داده شده، در نهایت به پایتون تبدیل شده است.

## پایگاه داده

برای ذخیره‌سازی اطلاعات از پایگاه داده PostgreSQL استفاده شده است. این پایگاه‌داده یکی از بهترین و قدرتمندترین پایگاه‌داده‌های رابطه‌ای می‌باشد که وظیفه ذخیره‌سازی اطلاعات پروژه را برعهده دارد، همچنین یکی از علت‌های انتخاب این پایگاه‌داده سازگاری بسیار بالای آن با پایتون اشاره نمود. اکثر اطلاعات این پروژه در قالب فایل‌های تصویری هستند که حجم این فایل‌ها بسیار بالا می‌باشد، لذا برای جلوگیری از افزایش حجم پایگاه داده، فایل‌ها بر روی سرور FTP نگهداری می‌شود و آدرس فیزیکی فایل‌ها در پایگاه‌داده سرور ذخیره می‌شود. اطلاعات ذخیره شده در پایگاه داده شامل داده‌های اخذ شده از ایستگاه‌های هواشناسی و شبکه سنسورهای بی‌سیم به همراه مختصات جغرافیایی آنها و نیز نقشه حاصل از اجرای مدل پیش‌بینی بیماری می‌باشد.

## جانگو

برای اینکه بتوان به برنامه قابلیت اجرای تحت وب داد و برای آن رابط گرافیکی تعریف کرد، نیاز به فریم‌ورک جانگو است. جانگو یک چارچوب سطح بالا است که بسیاری از موارد برنامه‌نویسی را بصورت خودکار فراهم کرده و در اختیار برنامه نویس قرار می‌دهد. در نتیجه برنامه‌نویس نیازی به نوشتن کدهای اضافی ندارد. این فریم‌ورک حرفه‌ای بودن را در کنار سرعت به ارمغان می‌آورد. از جمله سایت‌های معروف که از پایتون و جانگو استفاده کرده‌اند می‌توان به موارد زیر اشاره کرد:

YouTube, Instagram, Mozilla Firefox, Pinterest, Spotify, Reddit, NASA.

سرور جانگو پلتفرمی مرکزی برای تمام پروژه‌های گروه سنجش از دور است. این پلتفرم بستری یکپارچه برای ایجاد و توسعه هر پروژه جدید است.

از جمله وظایف سرور جانگو می‌توان به موارد زیر اشاره کرد:

- جمع‌آوری درخواست‌ها.
- پردازش.
- ذخیره‌سازی نتایج بر روی پایگاه داده.
- ایجاد رابط گرافیکی.
- آماده‌سازی داده‌های مورد نیاز الگوریتم‌ها.

## پروتکل ارتباطی با سرورهای خارجی

برای ارتباط بین سرور پایتون و هر سرور دیگری مستقل از زبان پیاده‌سازی سرور یا سیستم عامل سرور پروتکل Django REST framework استفاده شده است و به این طریق ارتباط ماشین-ماشین فراهم می‌شود. این فریم-

ورک یک چارچوب قدرتمند و قابل انعطاف برای ساختن API‌های تحت وب است و از ویژگی‌های آن می‌توان به موارد زیر اشاره کرد:

- استفاده از سیستم احراز هویت OAuth1a و OAuth2.
- قابلیت سریال‌سازی داده‌های ORM و non-ORM.
- استفاده در شرکت‌های مشهور مثل Mozilla, Red Hat, Heroku و Eventbrite.

### الگوریتم

این الگوریتم در حوزه پیش‌بینی و پیش‌هشدار شیوع بیماری فوزاریم کاربرد دارد. خروجی این الگوریتم نقشه پیش‌بینی میزان شدت بیماری است که توسط محاسبات مدل ارائه شده در الگوریتم بدست آمده است. این الگوریتم در مرکزی‌ترین بخش قرار گرفته است و توسط کارشناسان فنی گروه سنجش از دور توسعه داده شده است و در نهایت به زبان پایتون تبدیل شده است.

### سرور ذخیره سازی تصاویر

به دلیل بالا بودن حجم فایل‌های تولید شده توسط الگوریتم‌ها و یا ورودی‌های آنها نیاز به یک سرور مستقل برای نگهداری داده‌ها می‌باشد. از این رو برای کاهش بار سرور اصلی تصمیم به ایجاد یک فایل سرور مستقل گرفته شد. فایل‌هایی که بر روی این سرور قرار می‌گیرند از طریق آدرس‌های فیزیکی که در داخل پایگاه‌داده وجود دارد توسط برنامه مورد استفاده قرار می‌گیرند.

## وب سایت هواشناسی

داده‌های مورد نیاز شامل پارامترهای هواشناسی از قبیل دما، رطوبت نسبی و میزان بارش است که به صورت آنلاین از وب سایت هواشناسی <https://www.accuweather.com/> استخراج می‌شود. داده‌های مورد نیاز با استفاده از API ارائه شده در این وب سایت قابل دسترسی هستند. این داده‌ها بعنوان یکی از ورودی‌های مورد نیاز، از طریق پروتکل REST به الگوریتم داده می‌شوند.

## سرورهای خارجی سرویس گیرنده

سرورهای خارجی سرویس گیرنده شامل هر سروری می‌شود که نیاز به دریافت اطلاعات از سامانه پیش‌بینی بیماری فواریوم دارد. سرورهای خارجی با استفاده از پروتکل REST با سرور جانگو ارتباط برقرار می‌کنند. برای اتصال سرور جانگو با هر سرور دیگری و همچنین برای دریافت و ارسال اطلاعات از پروتکل REST استفاده شده است. به این صورت بدون وابستگی به زبان برنامه‌نویسی یا نوع سرور قادر به ارتباط با هر سرور خارجی خواهد بود.

## vegetationdiseases

Show/Hide | List Operations | Expand Operations

GET /vegetationdiseases/api/fusarium/

### Response Messages

HTTP Status Code	Reason	Response Model	Headers
200			

Try it out!

GET /vegetationdiseases/api/fusarium/{id}/

### Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	(required)		path	string

### Response Messages

HTTP Status Code	Reason	Response Model	Headers
200			

Try it out!

GET /vegetationdiseases/api/region/

### Response Messages

HTTP Status Code	Reason	Response Model	Headers
200			

Try it out!

GET /vegetationdiseases/api/region/{id}/

### Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	(required)		path	string

### Response Messages

HTTP Status Code	Reason	Response Model	Headers
200			

Try it out!

شکل ۲-۱ پرتال ارتباطی با سرورهای خارجی

Django REST framework

pd

Region List Api

## Region List Api

OPTIONS

GET

GET /cropdetection/api/region/

HTTP 200 OK

Allow: GET, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
[
  {
    "id": 1,
    "name": "khorasanrazavi-bajestan",
    "last_modified_date": "2018-01-15T07:09:48.768957Z"
  },
  {
    "id": 2,
    "name": "iran_ghazvin_ghazvin",
    "last_modified_date": "2018-01-20T05:38:48.635408Z"
  },
  {
    "id": 3,
    "name": "iran_ardebil_ardebil",
    "last_modified_date": "2018-01-22T11:00:52.235178Z"
  },
  {
    "id": 4,
    "name": "iran_khorasanrazavi_bajestan",
    "last_modified_date": "2018-01-29T08:47:13.825670Z"
  },
  {
    "id": 5,
    "name": "iran_khorasanrazavi_firozeh",
    "last_modified_date": "2018-01-29T08:59:47.125106Z"
  },
  {
    "id": 6,
    "name": "iran_khorasanrazavi_jovain",
    "last_modified_date": "2018-02-03T06:04:08.774673Z"
  }
]
```

شکل ۱-۳ فرمت جیسون مربوط به مناطق برای ارتباط با سرورهای خارجی

## رابط کاربری

رابط کاربری، در واقع یک واسطه گرافیکی تحت وب بین کاربر و پایگاه داده است که امکان استفاده از پایگاه داده را برای کاربر فراهم می‌کند. کاربر در اینجا مدیر سایت است. این رابط گرافیکی بالاترین سطح دسترسی به کاربر را برای هر گونه حذف و اضافه و یا تغییر دادن همه داده‌های موجود در پایگاه داده می‌دهد. این رابط کاربری تمام پروژه‌هایی که در این سیستم طراحی و پیاده‌سازی می‌شود را به صورت یکپارچه و متمرکز نشان می‌دهد.

### Django administration

#### Site administration

VEGETATIONDISEASES	
Fusariums	<a href="#">+ Add</a> <a href="#">Change</a>
Regions	<a href="#">+ Add</a> <a href="#">Change</a>

#### Recent actions

**My actions**

- [+ BiomassLai object](#)  
Biomass lai
- [Fusarium object](#)  
Fusarium
- [+ Fusarium object](#)  
Fusarium
- [+ ghazvin](#)  
Region
- [+ Analyze](#)
- [iran\\_khorasanrazavi\\_jovain](#)  
Region
- [Analyze](#)
- [54695458-e873-4ba2-871f-689af9fca384](#)  
Evapotranspiration
- [+ 54695458-e873-4ba2-871f-689af9fca384](#)  
Evapotranspiration
- [+ ghazvin](#)  
Region

شکل ۱-۴ صفحه مدیریت پروژه پیش‌بینی بیماری فوزاریوم گندم





## Django administration

Home › Vegetationdiseases › Fusariums › Add fusarium

### Add fusarium

Region:

Result:

No file selected.  
result file(.tif).

Data:

null

Latlong:

null

شکل ۵-۱ رابط کاربری بین مدیر و پایگاه داده

### سیستم احراز هویت

سیستم احراز هویت برای تعریف کاربران با سطوح دسترسی مختلف در نظر گرفته شده است. بالاترین سطح دسترسی مربوط به مدیر سایت است که با استفاده از صفحه مدیریت می‌تواند به طور مستقیم تمام داده‌های داخل پایگاه داده را مدیریت کند.

## Django administration

Home » Authentication and Authorization » Users » pd

### Change user

Username:

pd

Required. 150 characters or fewer. Letters, digits and @/./+/-/\_ only.

Password:

algorithm: pbkdf2\_sha256  
iterations: 36000

salt: AZxWxq\*\*\*\*\*

hash: dp4bJp\*\*\*\*\*

Raw passwords are not stored, so there is no way to see this user's password, but you can change the password using [this form](#).

#### Personal info

First name:

Last name:

Email address:

#### Permissions

☒ Active

Designates whether this user should be treated as active. Unselect this instead of deleting accounts.

☒ Staff status

Designates whether the user can log into this admin site.

☒ Superuser status

Designates that this user has all permissions without explicitly assigning them.

Groups:

Available groups ?

Q Filter

Choose all ?

Chosen groups ?

Remove all

The groups this user belongs to. A user will get all permissions granted to each of their groups. Hold down "Control", or "Command" on a Mac, to select more than one.

User permissions:

Available user permissions ?

Q Filter

admin | log entry | Can add log entry  
admin | log entry | Can change log entry  
admin | log entry | Can delete log entry  
auth | group | Can add group  
auth | group | Can change group

Chosen user permissions ?

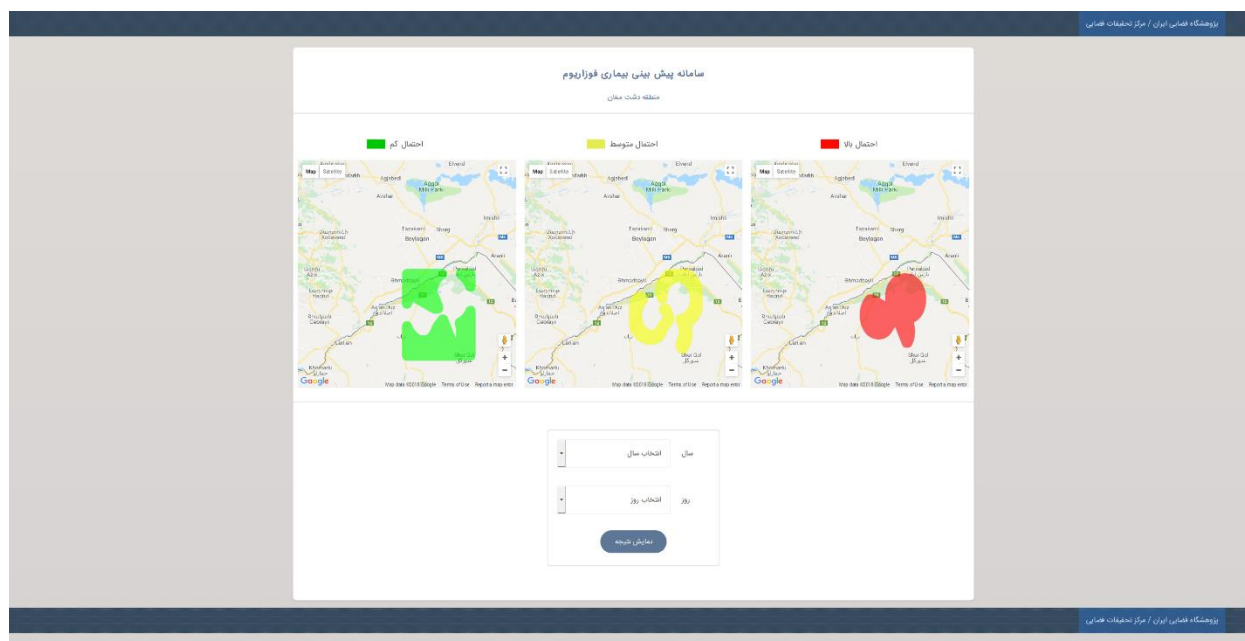
شکل ۱-۶ سیستم احراز هویت سامانه پیش‌بینی بیماری فوزاریوم گندم

## مدیر سیستم

مدیر سایت با بالاترین سطح دسترسی با استفاده از سیستم احراز هویت به صفحه مدیریت متصل می‌شود.

## خروجی‌های پردازش شده

پس از اجرای الگوریتم، خروجی تحت وب آن به شکل زیر خواهد بود. کاربر می‌تواند با انتخاب تاریخ مورد نظر اطلاعات پردازش شده را مشاهده نماید.



شکل ۷-۱ نقشه حاصل از پیش‌بینی احتمال وقوع بیماری

## ساختار کدها

در این بخش به معرفی و شرح ساختار کدها، ماژول‌ها و کلاس‌ها پرداخته خواهد شد.

```

.
├── manage.py
├── media
│   └── VegetationQuantitativeParameterApp
│       └── Regions
│           ├── ghazvin
│           │   └── Analyzes
│           │       └── 137bd913-564d-4f2e-b3c7-8b197a3f25c2
│           └── moghan
│               └── Analyzes
│                   └── 475e280d-4e7d-4ddc-a46f-d7446a62a515
├── QuantitativeParameterWeb
│   ├── __init__.py
│   ├── settings.py
│   ├── urls.py
│   └── wsgi.py
├── templates
│   ├── index.js
│   ├── registration
│   │   └── login.html
│   └── stylelogin.css
└── vegetationquantitativeparameter
    ├── admin.py
    ├── api
    │   ├── __init__.py
    │   ├── serializers.py
    │   ├── urls.py
    │   └── views.py
    ├── apps.py
    ├── __init__.py
    └── lib
        ├── biomass_lai
        │   ├── ghazvin.py
        │   ├── index
        │   │   ├── base.py
        │   │   ├── index.py
        │   │   └── __init__.py
        │   ├── __init__.py
        │   └── moghan.py
        ├── engine.py
        ├── evapotranspiration
        │   ├── example.py
        │   ├── files_reader.py
        │   ├── GeO.py
        │   ├── __init__.py
        │   ├── load_data.py
        │   ├── main_ET0.py
        │   ├── main_metric.py
        │   └── main_sebal.py
    
```

```

├── RasTerio.py
├── rs_mathematica.py
├── skl.py
├── test_metric.py
├── test_sebal.py
├── __init__.py
├── machinlearning
│   ├── algs
│   │   ├── algorithm.py
│   │   └── __init__.py
│   ├── __init__.py
│   └── main.py
├── migrations
│   ├── 0001_initial.py
│   ├── 0002_auto_20180819_0431.py
│   ├── 0003_auto_20180904_0716.py
│   └── __init__.py
├── models.py
├── static
│   └── vegetationquantitativeparameter
│       ├── css
│       │   ├── bootstrap-magnify.css
│       │   ├── bootstrap.min.css
│       │   ├── bootstrap-rtl.min.css
│       │   ├── font-awesome.min.css
│       │   ├── font-awesome.mins.css
│       │   ├── iranmap.css
│       │   ├── login.css
│       │   ├── main.css
│       │   ├── noscript.css
│       │   ├── owl.carousel.min.css
│       │   ├── owl.theme.default.min.css
│       │   └── style.css
│       ├── cssland
│       │   ├── font-awesome.mins.css
│       │   ├── images
│       │   │   ├── dark-bl.svg
│       │   │   ├── dark-br.svg
│       │   │   ├── dark-tl.svg
│       │   │   ├── dark-tr.svg
│       │   │   ├── light-bl.svg
│       │   │   ├── light-br.svg
│       │   │   ├── light-tl.svg
│       │   │   └── light-tr.svg
│       │   └── overlay.png
│       ├── main.css
│       └── noscript.css
│   ├── csslogin
│   │   ├── index.js
│   │   └── stylelogin.css
│   └── fonts
│       └── eot

```

```

— IRANSans (FaNum) _Medium.eot
— IRANSansWeb (FaNum) _Bold.eot
— IRANSansWeb (FaNum) .eot
— IRANSansWeb (FaNum) _Light.eot
— IRANSansWeb (FaNum) _Medium.eot
— IRANSansWeb (FaNum) _UltraLight.eot
— FontAwesome.otf
— fontawesome-webfont.eot
— fontawesome-webfont.svg
— fontawesome-webfont.ttf
— fontawesome-webfont.woff
— fontawesome-webfont.woff2
— iransans
  — eot
    — IRANSans (FaNum) _Medium.eot
    — IRANSansWeb (FaNum) _Bold.eot
    — IRANSansWeb (FaNum) .eot
    — IRANSansWeb (FaNum) _Light.eot
    — IRANSansWeb (FaNum) _Medium.eot
    — IRANSansWeb (FaNum) _UltraLight.eot
  — ttf
    — IRANSansWeb (FaNum) _Bold.ttf
    — IRANSansWeb (FaNum) _Light.ttf
    — IRANSansWeb (FaNum) _Medium.ttf
    — IRANSansWeb (FaNum) .ttf
    — IRANSansWeb (FaNum) _UltraLight.ttf
  — woff
    — IRANSansWeb (FaNum) _Bold.woff
    — IRANSansWeb (FaNum) _Light.woff
    — IRANSansWeb (FaNum) _Medium.woff
    — IRANSansWeb (FaNum) _UltraLight.woff
    — IRANSansWeb (FaNum) .woff
  — woff2
    — IRANSansWeb (FaNum) _Bold.woff2
    — IRANSansWeb (FaNum) _Light.woff2
    — IRANSansWeb (FaNum) _Medium.woff2
    — IRANSansWeb (FaNum) _UltraLight.woff2
    — IRANSansWeb (FaNum) .woff2
— TitrWeb.woff
— ttf
  — IRANSansWeb (FaNum) _Bold.ttf
  — IRANSansWeb (FaNum) _Light.ttf
  — IRANSansWeb (FaNum) _Medium.ttf
  — IRANSansWeb (FaNum) .ttf
  — IRANSansWeb (FaNum) _UltraLight.ttf
— woff
  — IRANSansWeb (FaNum) _Bold.woff
  — IRANSansWeb (FaNum) _Light.woff
  — IRANSansWeb (FaNum) _Medium.woff
  — IRANSansWeb (FaNum) _UltraLight.woff
  — IRANSansWeb (FaNum) .woff
— woff2

```

```

├── IRANSansWeb (FaNum) _Bold.woff2
├── IRANSansWeb (FaNum) _Light.woff2
├── IRANSansWeb (FaNum) _Medium.woff2
├── IRANSansWeb (FaNum) _UltraLight.woff2
├── IRANSansWeb (FaNum) .woff2
├── images
│   ├── 3-4.jpg
│   ├── back.png
│   ├── cd-top-arrow.svg
│   ├── favicon.ico
│   ├── font-mask.png
│   ├── image-1.jpg
│   ├── infohome.png
│   ├── info-section.jpg
│   ├── landing-bg-big-3.jpg
│   ├── leaf.png
│   ├── line.svg
│   ├── logo.png
│   ├── photo.jpg
│   ├── svg.png
│   ├── UserAvator.png
│   └── water-drops.png
├── js
│   ├── animatescroll.min.js
│   ├── _back-to-top.js
│   ├── bootstrap-magnify.min.js
│   ├── bootstrap.min.js
│   ├── breakpoints.min.js
│   ├── browser.min.js
│   ├── Chart.bundle.js
│   ├── _custombox.min.js
│   ├── custom.js
│   ├── index.js
│   ├── indexs.js
│   ├── iranmap.js
│   ├── jquery-3.2.1.min.js
│   ├── jquery.dropotron.min.js
│   ├── jquery.min.js
│   ├── jquery.scrollex.min.js
│   ├── jquery.scrolly.min.js
│   ├── _legacy.min.js
│   ├── main.js
│   ├── owl.carousel.js
│   ├── owl.carousel.min.js
│   ├── util.js
│   └── utils.js
└── templates
    └── vegetationquantitativeparameter
        ├── biomasslai_detail.html
        ├── biomasslai_list.html
        ├── et_lai_detail.html
        └── evapotranspiration_detail.html
    
```

```

— evapotranspiration_list.html
— home.html
— index.html
— land.html
— login.html
— region_detail.html
— region_list.html
— tests.py
— urls.py
— views.py
    
```

ساختار کلی ماژول‌ها به شرح زیر می‌باشد. در ادامه به شرح دقیق‌تر هر کدام می‌پردازیم.

نام پکیج	توصیف پکیج	نام ماژول	توصیف ماژول	نام کلاس	توصیف کلاس
Quantitative Parameter Web	این پکیج پروژه می‌باشد و شامل تنظیمات اصلی پروژه است	settings.py	تنظیمات اصلی پروژه برای وب	-	-
		Urls.py	تعریف تعداد تعداد urlهای مورد نیاز بر حسب پروژه و اتصال آنها به ماژول views.py	-	-
		wsgi.py	ماژول ارتباط با وب سرور.	-	-
		tests	برای تست و آزمون پروژه.	-	-
		urls	تعریف تعداد تعداد urlهای مورد نیاز بر حسب پروژه.	-	-
		views	واسط ارتباطی بین url پایگاه داده و قالب‌ها و الگوریتم‌های پیاده سازی شده.	-	-



Templates	حاوی قالب های , html css, js مورد نظر برای طراحی فرانت اند است.	-	-	-	-
static	حاوی فایل های استاتیک شامل css و تصاویر.	-	-	-	-
media	حاوی فایل هایی است که توسط الگوریتم تولید می- شود.	Regions	فایل های تولید شده به ازای هر منطقه دسته بندی می شوند.	-	-
vegetationquantitativeparameter	پکیج کلی پروژه، شامل الگوریتم های پیاده سازی شده.	admin.py	ماژول کنترل و ساخت صفحه ادمین، برای ارتباط مستقیم مدیر کل سیستم با پایگاه داده.	RegionAd min	صفحه ادمین مناطق.
				Evapotran spirationA dmin	صفحه ادمین پروژه Evapo Transpiration
				BiomassL aiAdmin	صفحه ادمین پروژه Biomass و Lai
		apps	برای اتصال به اپلیکشن سرور.	Vegetatio nquantitati veparamet erConfig	برای ایجاد تنظیمات پروژه با سرور.
		models	برای طراحی و پیاده سازی پایگاه داده.	Region	طراحی و پیاده سازی پایگاه داده مربوط به مناطق.
				Evapotran spiration	طراحی و پیاده سازی پایگاه داده مربوط به تحلیل های ET.
				BiomassL ai	طراحی و پیاده سازی پایگاه داده مربوط به تحلیل های LaiBiomass

		tests	برای تست و ازمون پروژه.	-	-
		urls	تعریف تعداد تعداد url های مورد نیاز بر حسب اپلیکشن .	-	-
		views	واسط ارتباطی بین url پایگاه داده و قالب ها و الگوریتم های پیاده سازی شده.	BiomassLaiListView	نمایش کل داده های BiomassLai.
				BiomassLaiDetailView	نمایش داده های BiomassLai به صورت جزئی.
				RegionListView	نمایش کل داده های مناطق.
				RegionDetailView	نمایش داده های مناطق به صورت جزئی.
				ETListView	نمایش کل داده های ET.
				ETDetailView	نمایش داده های ET به صورت جزئی.
api	پروتکل ارتباطی برای ماشین به ماشین.	serializers	ماژول سریال کردن داده های، پایگاه داده.	RegionSerializer	برای سریال کردن داده های مناطق از روی پایگاه داده.
				EvapotranspirationSerializer	برای سریال کردن داده های ET از روی پایگاه داده.

				BiomassLaiSerializer	برای سریال کردن داده های BiomassLai از روی پایگاه داده.
		urls	تعریف تعداد url های مورد نیاز بر حسب ارتباطات بین ماشین ها و اتصال آنها به ماژول views.py	-	-
		views	واسط ارتباطی بین url پایگاه داده و الگوریتم های پیاده سازی شده.	RegionListAPIView	واسط ارتباطی بین url پایگاه داده برای مشاهده مناطق.
				RegionRetrieveAPIView	واسط ارتباطی بین url پایگاه داده و مناطق، برای تغییر اطلاعات.
				EvapotranspirationListAPIView	واسط ارتباطی بین url پایگاه داده برای مشاهده تحلیل های ET.
				EvapotranspirationRetrieveAPIView	واسط ارتباطی بین url پایگاه داده و مناطق، برای تغییر اطلاعات تحلیل های ET.
				EvapotranspirationRetrieveAPIRun	واسط ارتباطی بین url پایگاه داده و مناطق برای اجرای الگوریتم ET.

				BiomassLaiListApiView	واسط ارتباطی بین url پایگاه داده برای مشاهده تحلیل های BiomassLai.
				BiomassLaiRetrieveAPIView	واسط ارتباطی بین url پایگاه داده و مناطق، برای تغییر اطلاعات تحلیل های BiomassLai
				BiomassLaiRetrieveAPIRun	واسط ارتباطی بین url پایگاه داده و مناطق برای اجرای الگوریتم BiomassLai
migrations	پکیج نگهداری و یکپارچه سازی تغییرات روی پایگاه داده.	-	-	-	-
lib	پکیج حاوی الگوریتم های سنجنش از دور توسعه داده شده در پروژه.	engine	واسط ارتباطی بین لایه view و الگوریتم های توسعه داده شده.	-	-
		biomass_lai	پکیج توسعه داده شده برای الگوریتم biomass.	-	-
		evapotranspiration	پکیج توسعه داده شده برای الگوریتم evapotranspiration.	-	-
		machinelearning	پکیج تحلیل داده و یادگیری ماشین برای تحلیل داده.	-	-
biomass_lai	پکیج مربوط به الگوریتم های biomass_lai.	ghazvin	الگوریتم های biomass_lai مربوط به منطقه قزوین.	Maize	الگوریتم های biomass_lai مربوط به محصول Maize.

				Alfalfa	الگوریتم های biomass_lai مربوط به محصول Alfalfa .
				Ghazvin	الگوریتم های منطقه قزوین.
		moghan	الگوریتم های biomass_lai مربوط به منطقه مغان.	Maize	الگوریتم های biomass_lai مربوط به محصول Maize .
				Alfalfa	الگوریتم های biomass_lai مربوط به محصول Alfalfa .
				Moghan	الگوریتم های منطقه مغان.
	evapotranspiration	files_reader	ماژول مربوط به کار با فایل با فرمت های مختلف.	-	-
		load_data	ماژول بارگزار، برای لود کردن داده‌های مورد نیاز الگوریتم.	Param	پارامترهای زمینی.
				META	پارامترهای متا دیتا.
				RasterBand	تبدیل کننده raster به ارایه numpy.
				Band	باندهای مختلف ماهواره.

# گزارش معماری سامانه پروژه شناسایی بیماری‌های گندم (فاز سوم پروژه)



				HotCold	Latitude و longitude پیکسل های گرم و سرد.
				Radiance	پارامترهای رادیانس.
				Weighting Coefficient t	وزن های مربوط به هر باند.
				Reflectanc e	پارامترهای Reflectance.
				Geographi calCoordi nates	طول و عرض جغرافیایی.
				Constants	مقادیر ثابت.
				Reflectanc eMetric	محاسبه کننده رفلکتانس به ازای باندهای مختلف.
				FileDir	آدرس فایل های مورد نیاز الگوریتم.
				۱۰ET_ _Params	پارامترهای مورد نیاز الگوریتم ET.
		main_ET0	الگوریتم ET0.	-	-

		main_metric	الگوریتم metric.	-	-
		main_sebal	الگوریتم sebal.	-	-
		RasTerio	ماژول اتصال و کار با کتابخانه rasterio.	-	-
		rs_mathematica	ماژول فرمول‌های مهندسی سنجش از دور.	-	-
		skl	ماژول یادگیری ماشین و تحلیل داده.	-	-
		test_metric	ماژول تست الگوریتم متریک.	-	-
		test_sebal	ماژول تست الگوریتم سبال.	-	-
machinlearning	پکیج الگوریتم‌های یادگیری ماشین	main	ماژول واسطه بین واسط تحت وب و الگوریتم یادگیری ماشین.		
		algs	پکیج حاوی الگوریتم‌های یادگیری ماشین.	algorithm	الگوریتم‌های random_forest و decesition_tree و gussian_process_r . svr و egresion

کد سند: MSRI- SAP9997-01-R-03/01

ویرایش: 1/1

طبقه‌بندی: عادی

گزارش معماری سامانه پروژه شناسایی بیماری‌های گندم  
(فاز سوم پروژه)



پیوست‌ها

پیوست ۱- کدهای پروژه