

## عنوان و اطلاعات کلی پروژه

- نام پروژه: سامانه مدیریت پایان نامه ها
- نام و نام خانوادگی: مهدیه قاسم زاده
- شماره دانشجویی: ۴۰۱۱۳۲۰۰۰۴۲
- کد ملی: ۴۰۶۱۴۶۷۱۰۷ (که در سایت اشتباها ۴۰۴۱۴۶۷۱۰۷ درج شده)
- آدرس ریپوزیتوری:

## خلاصه پروژه

- هدف پروژه: هدف از این پروژه، پیاده سازی یک سامانه جامع و کاربردی برای مدیریت تمامی مراحل فرآیند پایان نامه های دانشجویی، از آغاز درخواست تا ارزیابی نهایی است.
- توضیح کوتاه عملکرد کلی برنامه: سامانه از دو بخش اصلی دانشجو و استاد تشکیل شده است. دانشجو می تواند درخواست اخذ پایان نامه، مشاهده وضعیت، درخواست دفاع و جستجو در آرشیو را انجام دهد. استاد می تواند درخواست های دانشجویان را مدیریت کرده، نمرات را ثبت کند و در آرشیو پایان نامه ها جستجو نماید. کلیه داده ها در فایل های JSON ذخیره می شوند و برنامه از طریق رابط خط فرمان (CLI) اجرا می شود.

## نیازمندی ها

- نسخه پایتون Python: ۳.۱۳
- کتابخانه ها / پکیج های استفاده شده:

- **json**: برای ذخیره داده‌ها به عنوان پایگاه داده
- **os** و **datetime**: برای مدیریت تاریخ‌ها و مسیر فایل‌ها

- سایر نیازمندی‌ها:

- سیستم کنترل نسخه git
- فایل‌های داده‌ای با فرمت json
- فایل‌های پایان‌نامه با فرمت pdf

## ساختار پروژه

ساختار پروژه به صورت ماژولار و شی‌گرا طراحی شده است. هر ماژول مسئولیت خاصی را بر عهده دارد تا کد خوانایی و قابلیت نگهداری بالاتری داشته باشد.

- **main.py**: نقطه ورود برنامه. حاوی کلاس ThesisApp است که جریان اصلی برنامه و منوهای کاربری (دانشجو و استاد) را مدیریت می‌کند.
- **services/**: پوشه‌ای که ماژول‌های اصلی پروژه در آن قرار دارند.
  - **services/auth\_service.py**: حاوی کلاس AuthService برای مدیریت فرآیندهای ورود به سیستم و تغییر رمز عبور.
  - **services/student\_service.py**: حاوی کلاس StudentService برای مدیریت عملیات مربوط به دانشجو مانند درخواست اخذ پایان‌نامه و جستجو در آرشیو.
  - **services/professor\_service.py**: حاوی کلاس ProfessorService برای مدیریت عملیات مربوط به استاد، از جمله مدیریت درخواست‌ها و ثبت نمره.
  - **services/data\_service.py**: حاوی کلاس DataService که مسئولیت اصلی آن مدیریت خواندن و نوشتن داده‌ها در فایل‌های JSON است.

- **data/**: پوشه‌ای برای ذخیره فایل‌های داده‌ای
  - **data/students.json**: اطلاعات دانشجویان
  - **data/professors.json**: اطلاعات اساتید
  - **data/courses.json**: اطلاعات دروس پایان‌نامه
  - **data/theses.json**: اطلاعات پایان‌نامه‌ها و وضعیت آن‌ها
  - **data/files/**: پوشه‌ای برای ذخیره فایل‌های PDF پایان‌نامه‌ها

## توضیح کلاس‌ها و توابع

- **main.py** در **ThesisApp**
  - توضیح عملکرد: کلاس اصلی برنامه که کل سیستم را اجرا و مدیریت می‌کند
  - متدها: `run()`, `student_menu()`, `professor_menu()`
- **AuthService** در **auth\_service.py**
  - توضیح عملکرد: مسئولیت احراز هویت و مدیریت کاربران
  - متدها (ایستا): `login()`, `change_password()`
- **StudentService** در **student\_service.py**
  - توضیح عملکرد: مدیریت تمام فعالیت‌های مربوط به دانشجو در فرآیند پایان‌نامه
  - متدها (ایستا): `request_thesis()`, `view_thesis_status()`, `request_defense()`, `search_theses_archive()`
- **ProfessorService** در **professor\_service.py**
  - توضیح عملکرد: مدیریت تمام فعالیت‌های مربوط به استاد در فرآیند پایان‌نامه

- متدها (ایستا): `manage_thesis_requests()`, `manage_defense_requests()`, `record_score()`

- **DataService در `data_service.py`:**

- توضیح عملکرد: مسئولیت اصلی مدیریت ذخیره سازی داده ها و دسترسی به اطلاعات کاربران

- متدها (ایستا): `load_data()`, `save_data()`, `get_user_name()`

## جزئیات پیاده سازی

- الگوریتم ها:

- جستجو در آرشیو: از یک الگوریتم جستجوی خطی برای یافتن پایان نامه ها بر اساس کلمه کلیدی در فیلدهای مختلف (عنوان، چکیده، کلمات کلیدی، نام استاد و دانشجو) استفاده شده است.

- نحوه اجرا

۱. تمامی فایل های پروژه را در ساختار پوشه ای مناسب قرار دهید.

۲. از طریق ترمینال، به پوشه اصلی پروژه بروید.

۳. دستور زیر را اجرا کنید: `python main.py`

- مراحل اجرای پروژه در سیستم کاربر

۱. برنامه با نمایش منوی اصلی آغاز می شود.

۲. کاربر نوع کاربری خود (دانشجو یا استاد) را انتخاب می کند.

۳. سیستم با دریافت کد کاربری و رمز عبور، اعتبار کاربر را بررسی می کند.

۴. پس از ورود موفق، کاربر به منوی مربوط به خود هدایت می شود و می تواند از امکانات برنامه استفاده کند.

## نمونه خروجی

این بخش یک نمونه از خروجی برنامه را نمایش می‌دهد:

○ بخش ورود و منوی اصلی

```
سامانه مدیریت پایان‌نامه‌ها

--- منوی اصلی ---
آیا دانشجو هستید یا استاد؟ (د/ا): د
5001: کد دانشجویی
رمز عبور: 123456
!ورود با موفقیت انجام شد. خوش آمدید، سارا محمدی

--- منوی دانشجو ---
1. درخواست اخذ پایان نامه
2. مشاهده وضعیت پایان نامه
3. درخواست دفاع
4. جستجو در آرشیو پایان نامه‌ها
5. تغییر رمز عبور
6. خروج
| :انتخاب
```

○ جست و جوی پایان نامه

```
عبارت جستجو را وارد کنید: هوش مصنوعی

نتایج یافت شده:
-----
عنوان: بررسی روش‌های نوین در یادگیری عمیق
چکیده: این پایان‌نامه به بررسی و مقایسه روش‌های پیشرفته در حوزه یادگیری عمیق و کاربردهای آنها می‌پردازد.
را دانلود کنید PDF متن: برای مشاهده متن کامل، فایل
کلمات کلیدی: یادگیری عمیق، شبکه عصبی، هوش مصنوعی
نویسنده: لیلا شهبازی
سال/نیمسال: 1404 / دوم
داوران: محمد احمدی، فرهاد مرادی
استاد راهنما: حسین ابراهیمی
data/files/thesis_T004.pdf: لینک دانلود فایل
نمره: 18.5 (الف)
-----
عنوان: طراحی سیستم تشخیص اخبار جعلی با الگوریتم های یادگیری ماشین
چکیده: این پایان نامه به امکان سنجی طراحی سیستم تشخیص اخبار جعلی با الگوریتم های یادگیری ماشین می پردازد.
را دانلود کنید PDF متن: برای مشاهده متن کامل، فایل
کلمات کلیدی: یادگیری ماشین، اخبار جعلی، هوش مصنوعی
نویسنده: زهرا رحمانی
سال/نیمسال: 1404 / دوم
```

## مشکلات و پیشنهادات

- ایده‌ها و عملکردهای پیاده‌سازی نشده:

- هش کردن رمز عبور: برای افزایش امنیت، رمز عبور کاربران باید قبل از ذخیره در فایل، هش شود که به دلیل محدودیت زمانی پیاده‌سازی نشد.
- سیستم نوتیفیکیشن: پیاده‌سازی سیستم ارسال پیام و نوتیفیکیشن بین کاربران که به دلیل محدودیت زمانی پیاده‌سازی نشد.
- ساخت رابط کاربری: با استفاده از کتابخانه Tkinter بخش زیادی از آن پیاده‌سازی شد که در فایل اصلی با نام gui.py ذخیره شده است اما بدلیل محدودیت زمانی و آغاز امتحانات پایان ترم دانشگاه موفق به کامل کردن آن نشدم.