Project 2-
A Full-Stack Django Web Site Project

Designed and Prepared by Nasreddine Hallam

TABLE OF CONTENTS

1. Project Overview

In this project, you will apply Python and Django coding knowledge and skills to create a fully featured web project.

The project encompasses four cooperating Django applications and provides visitors and member users with a web platform to communicate among themselves and manage Items (add, delete, change, rate, add comments and evaluate).

Students will work in groups of four members to create the web project. They must use a Version Control System (such as GitLab repository) to track the source code and collaborate among themselves.

2. Learning Objectives

- Evaluate requirements and translate them into a web project designs
- Use Django MVT (MVC) architecture to implement a server-side web project
- Deploy a web project to a production server
- Model data models using ORM (object Relational Mapping)
- Use templating to visualize the web applications
- Use a database to persist transactional data
- Develop and provide your own web site API web services
- Use a rest framework to serialize/deserialize data model into JSON

- Code self-contained and loosely coupled web applications
- Design the whole project into modules
- Use OOP approach to code modules
- Use a VCS (git) to collaborate with peers

3. Development/coding tools

This web project is a good exercise of a full-stack web development.

- Client-side:
  - HTML: use templating and inheritance.
  - CSS: use Flex or Grid layout
  - JavaScript. Use it for client-side validation and also coding some functionalities of your choice.
- Server-side
  - scripting: Django python
  - server-side database: PostgreSQL
  - deploy to the production server Heroku[1]

[1] A graded lab assignment will show how to deploy a Django Application to Heroku.

4. Technical Requirements

In this specification the project is also referred to as the website. Your project consists **of FOUR** self-contained and loosely coupled **Django applications,** namely the Administration app, the User Management app, the Messaging app and the Item Catalog management app. These applications cooperate with each other to fulfill the main functionality of the website. They are thoroughly explain in the next section.

3.1 Website Administration Application

This application is the dashboard of the whole website that is used by a superuser to manage four specific groups of users namely, **Admin_super_grp**, **Admin_user_grp**, **Admin_Item_gp**, and **Members.** These groups and their users are described below.

a. **Members**–a group that consists of (client) users who are already registered with the web project. Visitors are automatically added to **members** group once they register to the website.

b. **Admin_user_grp-** a group that consists of two users that only a superuser can set**.** Members of this group are responsible to access/manage users from group **members (defined above).** They are granted with privileges to:
   - add/delete/block/warn ... users in **Members** group.
   - Please set one user in this group with the following credentials: user: **user_manager1**, passwd: 456

c. **Admin_Item_grp**- a group that consists of two users that only a superuser can set**.** Members of this group are responsible to access/manage items. They are granted with privileges to:
   - add/delete/change items.
   - Please set a user of this group with the following credentials: user: **item_manager1**, passwd: 789

d. **Admin_super_grp**: a member of this group can manage all registered users (client members and the above two admin group members as well), items, and all other interactions:
   - A user of this group is called superuser.
   - add/delete/change/block/flag/warn... member users and the above two admin group users as well.
   - Add (but not delete or change) other new peer members to admin_super_gp.
   - Each admin operation should be archived in a log
   - Initially, create a superuser: nasr, with passwd: 123 as a member of this group.

3.2 Website User Management Application

This application is to manage the Member users and visitors.

1. a visitor is a non-registered user who can:
   a. access and browse some of the website content
   b. choose to register and obtain an account (in which case this user is added to **Members** group automatically)

2. a member is a registered user with an account with the following capabilities:
    a. login/logout
    b. reset password, …
    c. has a profile that consists of a name, email, and an avatar picture. They can update their profile.
    d. Can see notifications (e.g., warned, flagged, new messages)

Recall that admin users in the group **Admin_user_grp** can also

3. add new users
4. flag/warn any user

3.3 Website Messaging Application

This application deals with the communication between **Member** users only.

1. **Member** users can send private messages between themselves
2. **Member** users should be notified they have new messages upon login.
3. **Member** users must be able to comment on an item in the Catalog application.

3.4 The Website Item Catalog Application

This is the core and the primary application of the whole project. Much of user interactions such as operations and transactions are done through this application.

The Item word in this application is a generic concept. You will need to select what kind of item based on project themes you choose for your website. The main actors of this applications are **Member** users and visitors (non-registered users). Their roles in this core application are as follows:

5. Member users can add items
6. Member users (owners of the item) can delete/modify item
7. Member users can like/rate/flag an item that is not their own.
8. Member users can post a comment on any item
9. Member users can see all details of a given item that is not theirs

10. Users (both visitor and Member users) can browse/filter/search through existing items. This is the primary view of the website, which is the homepage. The list of items should be shown in paginated pages and visualized in different format, for e.g. grid or columns.

11. Through well-defined urls (and query strings), users can use web API to:
    a. list all items, specific items,
    b. list or a single item.
    c. update / delete items [your instructor may remove this requirement].
12. Users (visitors only) cannot display details of a single given item (they are routed to the registration page instead)

Recall that admin users in the group **Admin_Item_grp** can also

13. add new items
14. delete/modify/flag any item

5. Project Requirements

- Teams of four and three members **(appointed by your professor)**
- Follow software engineering best practices:
    - Proper classes and modules are a must in all your coding
    - inheritance (templates)
    - Must work with **Class-based** views
    - Can use function-based as well.
    - Use Django to its fullest when possible (for e.g., forms, authentication/authorization system, …)
- Must use the OOP methodology
    - Comments are required.
    - Naming, comments,
    - Where possible, use exception handling (try/except)
    - Applications must be self-contained and can easily be used in newer projects
    - A written report documenting the project development progress.

- Group members are expected to collaborate using a VCS. Specifically,
    - Use Git and Gitlab to manage code
    - Use the the feature branch (protected branch) model when developing features
    - Use Merge Requests to integrate feature (protected) branches
    - Ensure commit messages are meaningful and valuable
    - Perform code reviews as part of the Merge Request

Note- **individual marks** will be associated **with each group member**. Their contribution to the project will be measured using Git and Gitlab, so be sure to commit and work on code with your teammates.

6. Submission and deadline

1. Project duration (5weeks): **last day of class – 13th of May 2022**
2. Name project dw-42022-prj-grpX-ABCDEFG…
    a. **X** is a number that will be assigned to your group
    b. **ABCDEFG**… is the last name of the group leader
3. Must have **Readme** Text file detailing the content of your project and how to run your program(s); including the Heroku URL of your website.
4. Deploy to Heroku: see the lab Heroku document that will provided by your instructor on week14.
5. Submit a weekly report documenting the project development progress, the format is given in appendix.
6. **Submit a zipped file to Moodle**
    a. Only team leader can submit
    b. Your submission shouldn't include those files that are ignored in .ignore
    c. Include the Heroku URL of your website Feedback comment box.

7. Appendix

You may choose one project from the project themes given next.

6.1.1 Project Prototype Instance 1 – The academic project (item)

This is to show case projects (academic projects). Projects: are academic projects undertaken by students/researchers/ developers.

Academic projects can be of any type (practical, theoretical, fundamental research, empirical,…) and from any academic field (computing, science, medicine, social science, …)

A project is characterised by the following attributes:

- owner, name , type, field, keyword_list, description , url , status (completed, ongoing, planned,….), rate, a snapshot (relevant image(s)), …
- You may include other pertinent attributes if you deem it necessary.

6.1.2 Project Prototype Instance 2 – The documentary movie (item)

The website is to show allow visitors and members browse documentary movies.

Documentary movies can be of any genre (Expository, Essayistic, Observational, Participatory, research, Interview,…)

A documentary is characterised by the following attributes:

- Uploader(user), film-maker, title, genre, keyword_list, description ,
- url ,year_production, status (parent's guide, certificate,….), rate, a snapshot (relevant image(s)), …

You may include other pertinent attributes if you deem it necessary.

6.1.3 Project Prototype Instance 3 – The product (item)

The website is to allow visitors and members browse (sellable) products.

Products can be of any genre (book, cds, videos, games, electronic, furniture, baby stuffs, …)

A product is characterised by the following attributes:

- owner(user), title, genre, description , price, address,
- status(new , used, ….), rate, a snapshot (relevant image(s)), …

You may include other pertinent attributes if you deem it necessary.

6.1.4 Software Development Job Board

This website is designed to show case a series of jobs. A Job is made up of the following elements:

- · Title , Category (o E.g., Web dev, Mobile dev, Embedded dev, etc.
- Environment (E.g., Javascript, Java, C#, Python, etc.), Wage , Type (E.g. Full-time, part-time, contract), Keyword List (a list of keywords associated with the product),
- Description , Status (I.e., Available, Filled, No-longer exists), Banner image (An image representing the Job)

Additional elements can be added to create a complete item following this theme.

6.1.5 Common specs to all project themes.

Regardless of the project you select, the following requirements must be met by your website.

Visitors to the website

- Can browse all items, or a sub-set (filtering by owner, category, popularity,…).
- The list of items is paginated, and only title, owner, keyword list should be visible.
- **Cannot go** to the item detail page. They have to be registered first and then logged in.

Registered (Members) users can:

- browse items,
- go to the detail item page,
- add/delete/change their own items, and
- rate other items.
- comment on any item.
- chat with each other (send messages between themselves)

The superuser a user who is in the Admin_super_grp) can:

- create other users,
- assign them to **Admin_user_grp**, **Admin_Item_grp** and **Admin_super_grp** to allow them administer (items and/or users and/or messages)
- assign them to **members.**
- create/delete/update/warn/flag other users / assign them to any group
  Don't forget to add my name into **Admin_super_grp** (**nasr**, passwd **123**)

A user who is in the Admin_ user _grp)

- add/delete/block/warn ... users in **Members** group.
- Please set one user in this group with the following credentials:
  user: **user_manager1**, passwd: **456**

A uAdmin_ item_grp)

- add/delete/change items.
- Please set a user of this group with the following credentials:
  user: **item_manager1**, passwd: **789**

6.1.6 Prepare and Use Test data

The following data and models must be present in your final submission:

1. Ten Member users
2. Two Users in each admin group.
3. Ensure you have the special "Instructor" user as described above
4. Twenty items. Items must have different meta data, such as category, title, price, etc. based on the chosen theme
5. Rating, and comments should be present on some of the posts.
6. Some users should already have some messages between them in their inboxes

6.2 Progress Reports

A Written report documenting the project development progress must be submitted at the end of each week. These reports will be graded and contribute to the overall grade of the project. A table is provided as a sample of how the report can look. It must include the following information:

1. URLs to the merge requests representing the completed features, or on-going features of the week.
   - A summary of the overall status of the project
2. What tasks have been completed (with links to Merge Requests)
3. What tasks are schedule for the following weeks
4. Brief estimates of each task

At the end of each week, you (the leader) should submit a document in the format given below.

- The weekly summarized progress report are graded
- Document only major tasks

Week1 progress template

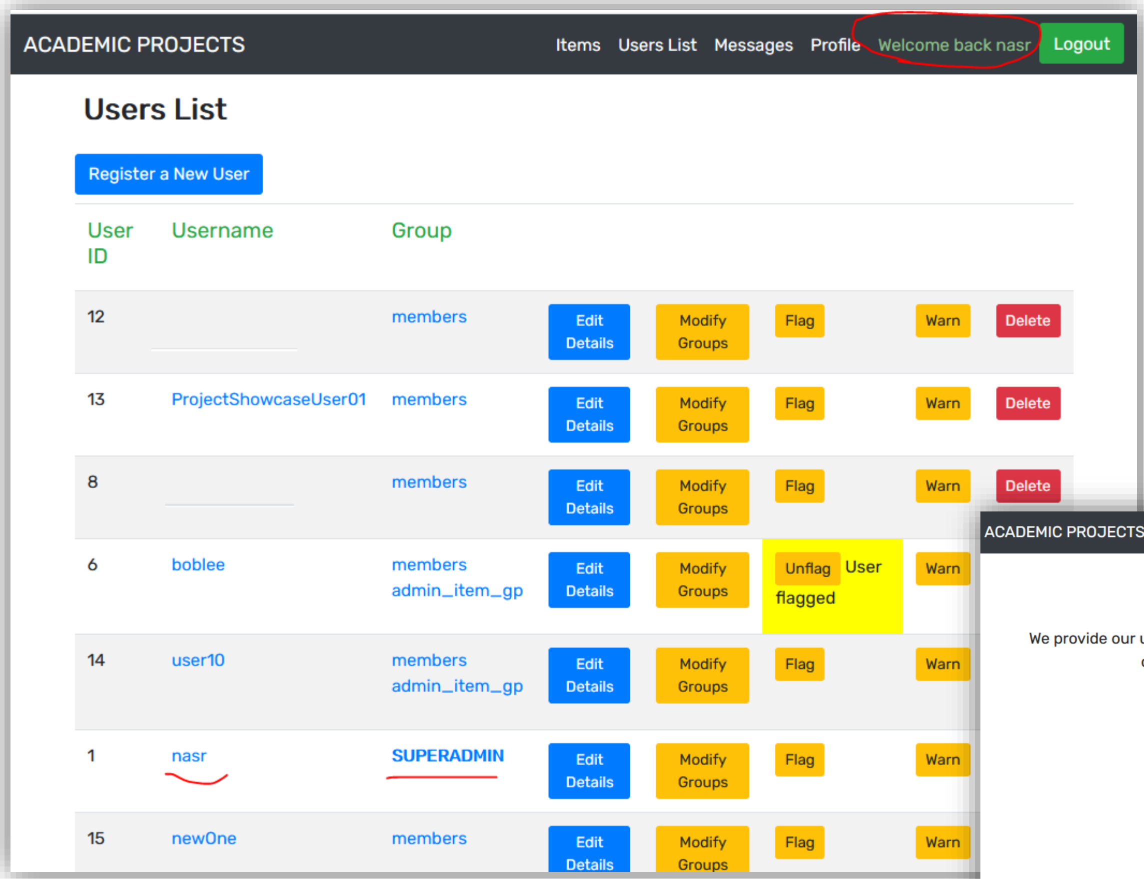
ID	Task	M.R. URL	Assignee	Date-start	Duration?	problems	Status	Note
1	Project init -meeting	Link …	all	W1-d1	2 hours	N/A	Completed 🟩🟩🟩🟩	closed
3	Admin app		badr	W1-d2	1 hour		25% 🟩⬜⬜⬜	Use shell to test
4	App1-models and shell		nasr	W1-d4	3 hours		50% 🟩🟩⬜⬜	
5	Items-models and shell		sara	W1-d4	4 hours		75% 🟩🟩🟩⬜	
. .	urls.py (app1)		..	…	…	…	…	…

Week2 progress template

ID	Task	M.R. URL	Assignee	Date-start	Dur	problems	Status	Note
T3	Admin app	Link …	Lim	W2-d2	3hours	50% 🟩🟩⬜⬜	All admin groups set	
T4	App1-models		Raj	W2-d1	2hours	50% 🟩🟩⬜⬜	T4	
T5	Items-models and shell		Diana	W1-d4	4hours	100% 🟩🟩🟩🟩	closed	
T3	Admin app		Lim	W2-d2	3hours	50% 🟩🟩⬜⬜	All admin groups set	

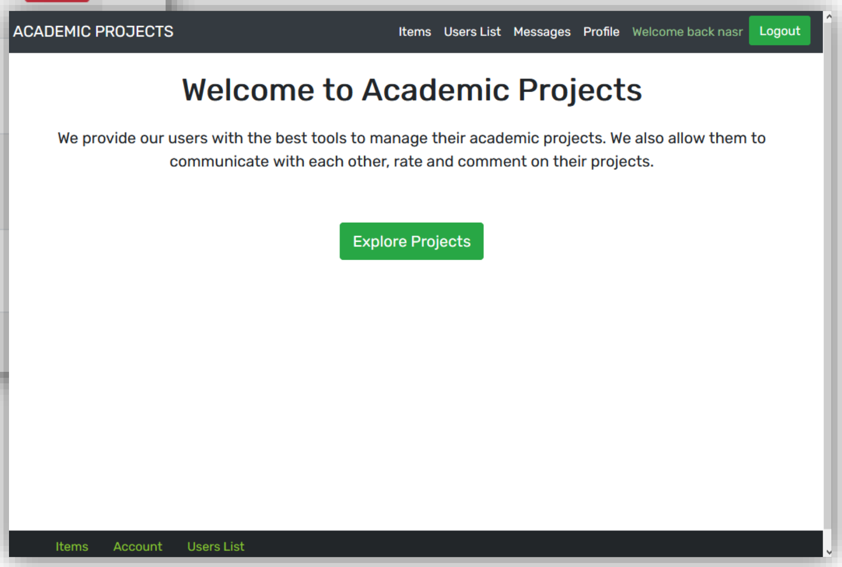
6.3 A sample of some self-explanatory snapshots

Below are some screenshots from some project



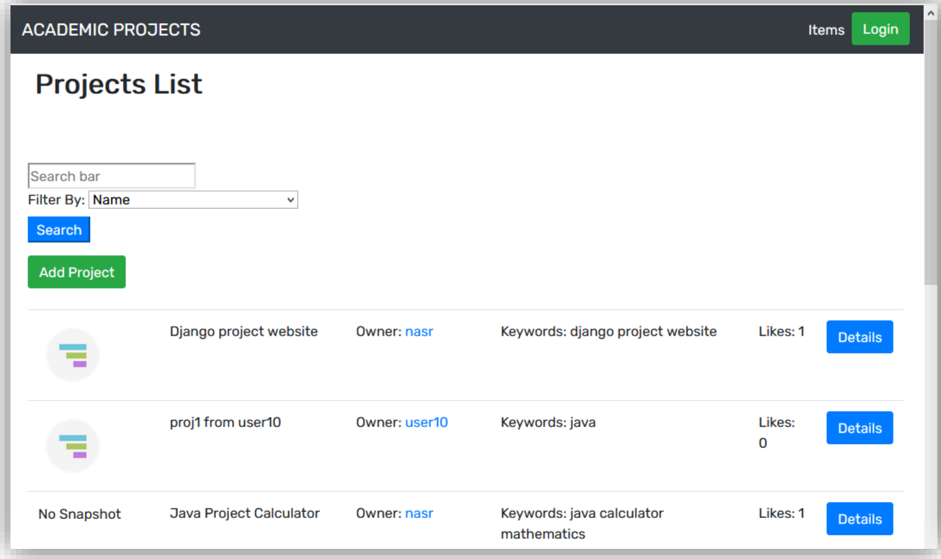
Users List

User ID	Username	Group					
12		members	Edit Details	Modify Groups	Flag	Warn	Delete
13	ProjectShowcaseUser01	members	Edit Details	Modify Groups	Flag	Warn	Delete
8		members	Edit Details	Modify Groups	Flag	Warn	Delete
6	boblee	members admin_item_gp	Edit Details	Modify Groups	Unflag User flagged	Warn	
14	user10	members admin_item_gp	Edit Details	Modify Groups	Flag	Warn	
1	nasr	**SUPERADMIN**	Edit Details	Modify Groups	Flag	Warn	
15	newOne	members	Edit Details	Modify Groups	Flag	Warn	



Welcome to Academic Projects

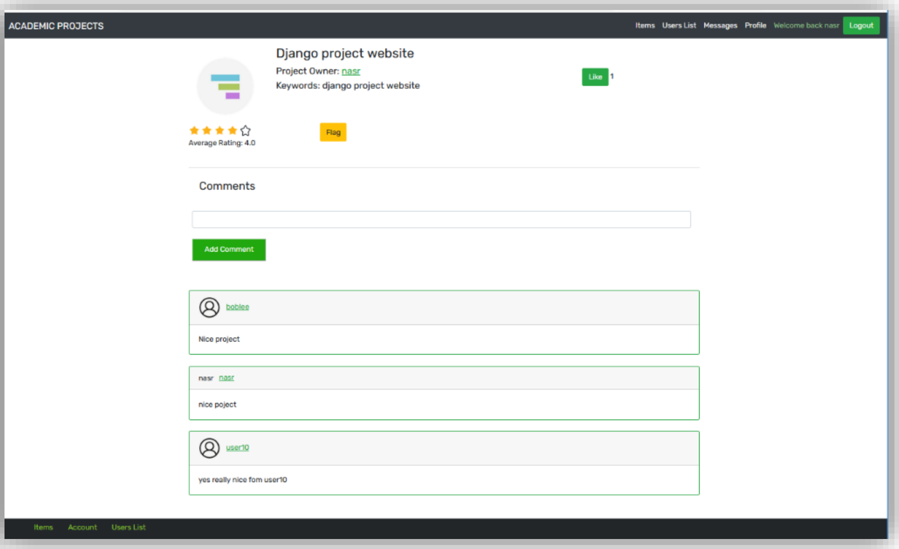
We provide our users with the best tools to manage their academic projects. We also allow them to communicate with each other, rate and comment on their projects.


ACADEMIC PROJECTS Items Login

Projects List

Search bar
Filter By: Name
Search
Add Project

	Django project website	Owner: nasr	Keywords: django project website	Likes: 1	Details
	proj1 from user10	Owner: user10	Keywords: java	Likes: 0	Details
No Snapshot	Java Project Calculator	Owner: nasr	Keywords: java calculator mathematics	Likes: 1	Details


ACADEMIC PROJECTS Items Users List Messages Profile Welcome back nasr Logout

Django project website
Project Owner: nasr
Keywords: django project website
Like 1
Average Rating: 4.0
Flag

Comments

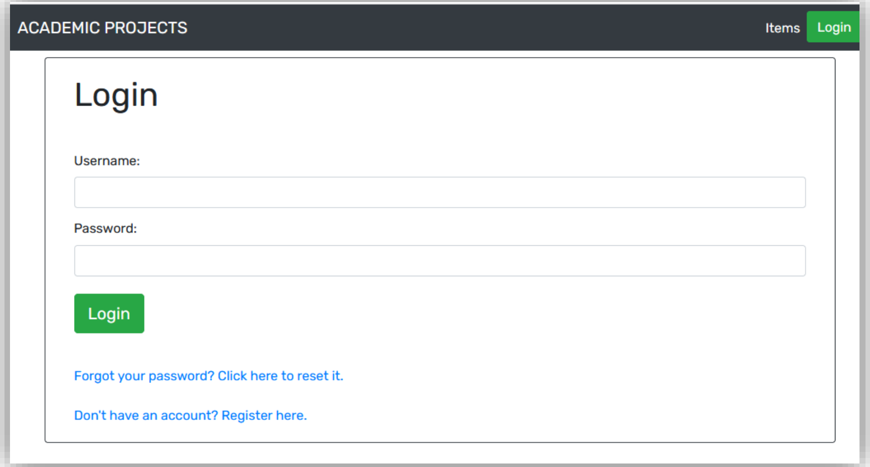
Add Comment

boblee
Nice project

nasr nasr
nice poject

user10
yes really nice fom user10

Items Account Users List


ACADEMIC PROJECTS Items Login

Login

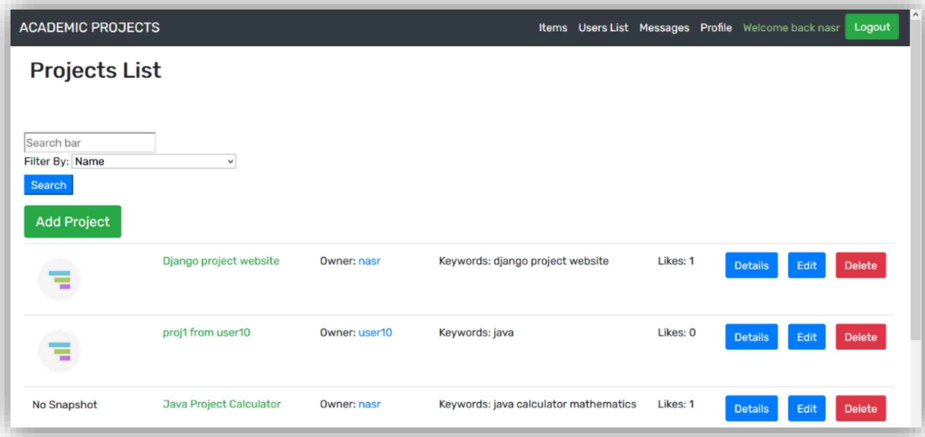
Username:

Password:

Login

Forgot your password? Click here to reset it.

Don't have an account? Register here.


ACADEMIC PROJECTS Items Users List Messages Profile Welcome back nasr Logout

Projects List

Search bar
Filter By: Name
Search
Add Project

	Django project website	Owner: nasr	Keywords: django project website	Likes: 1	Details	Edit	Delete
	proj1 from user10	Owner: user10	Keywords: java	Likes: 0	Details	Edit	Delete
No Snapshot	Java Project Calculator	Owner: nasr	Keywords: java calculator mathematics	Likes: 1	Details	Edit	Delete