1 INTRODUCTION

1.1 PURPOSE OF THE DOCUMENT

The purpose of this document is to present a description of the online grievance forum – sortmeout. It will explain the system's overall purpose and key features, interfaces, operations and design. This document is intended for both the stakeholders and the developers of the system.

1.2 MOTIVATION FOR THE PROJECT

The online forums already available today either commit themselves to specific domains or propose solutions that are not personalized based on the user's level of understanding of his problem/grievance.

Our first aim is to keep the project entirely user-centric i.e. the user must be in a position to express his problem and his level of its understanding fully. The user may be an expert in the field in which his grievance lies, or he may be a total newbie. Our aim is to give him a solution that exactly matches his expectations. The proposed solutions should be such that it can be well comprehended by the user and shouldn't be too complex for a novice or too basic for an expert.

Also, at sortmeout, a user can post a grievance to any category – including a special category "unlisted". This means that if sortmeout currently lacks the expertise to tender to the user's grievance, experts may be contacted and collaborated through external means by the moderators for serving the user.

1.3 SCOPE OF THE PROJECT

sortmeout is an online grievance forum for a wide range of grievance categories. It primarily facilitates a categorization of user grievances and walk-through resolution of the same through the medium of domain experts (preferably) and/or other Internet users (not necessarily part of the sortmeout userbase). Apart from the functional aspects this forum presents a solid yet easy-to-understand user interface.

More specifically, sortmeout is designed as a link between a user and experts, who expertise in the domain(s) of the user's grievance. It then allows the user to walk himself through to the resolution of his grievance through a path of solutions and interim grievances.

1.4 GLOSSARY OF KEY TERMS

Administrator	Site Administrator
Moderator	Person who reviews incoming grievances and solutions for inappropriate content
Expert	Site user who specializes in some domain(s) and has completed the registration as an expert
Registered user/ Member	Site user who has completed the registration as a normal member
Internet user/ Visitor	Site user who has not completed the registration process
Grievance	A member-originated description of his problem
Solution	A viable action intended to solve a member's grievance
Interim Grievance	A restatement of a member's partially-solved grievance
Closing a Grievance	Closing a grievance permanently by selecting a solution that best solves the grievance fully/partially (disallow selection and posting of more solutions to the grievance)

Finalizing a Grievance	Closing a grievance thread permanently by selecting a solution that completely solves the last interim grievance fully (disallow selection and posting of more solutions and interim grievances to the grievance thread)
Public forum	A view of all the grievances marked as "public" by the originators of the grievances
Private forum	A member's personal view of his self-originated grievances marked as "private"
Category	Domain of expertise that closely classifies a grievance

2 Overall Description

2.1 System Architecture

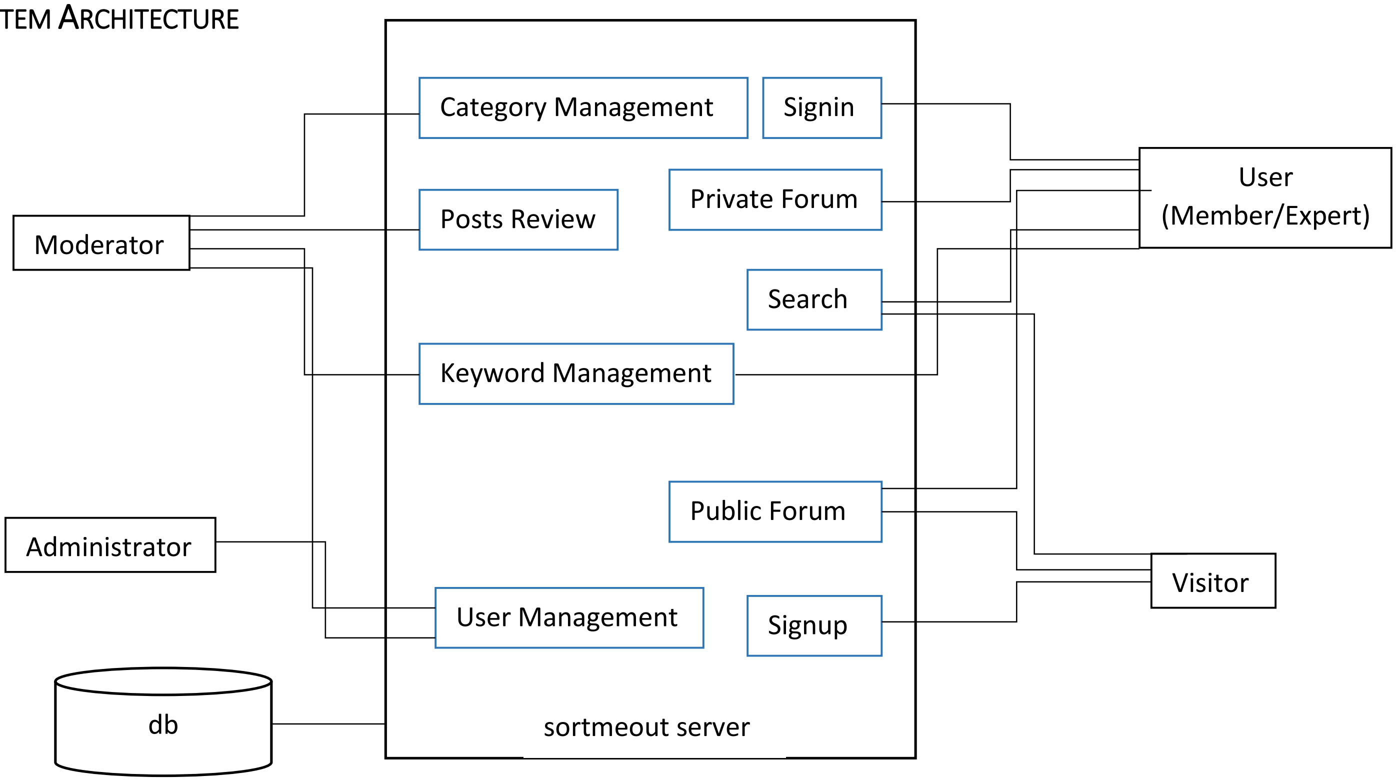


Figure 2.1 System block diagram

The sortmeout system has 3 major types of actors. Their names and key functionalities are as listed below.

1> **Administrator** – complete access to the User Management Subsystem

2> **Moderator** – access to the User Management and Forum Subsystems for validating purposes

3> **User**

- a. **Visitor** – can only post solutions in public forum
- b. **Member** – can post grievances in public and private forums; solutions in public forum
- c. **Expert** – can post grievances of their own in public and private forums and post solutions to even category-specific private forums of other Members/Experts

2.2 System Modules and Core Functionality

1> **User Management** (trivial; not covered in this document)– Member and Expert Signup, Signin

2> **Forums** – User's Private Forum, Public Forum (with and without search results), Expert Domain-Grievances View

3> **Review** (by configuring Django Admin Interface; not covered in this document) – Post Review, Expert Registration Review, New Category/Keyword Review/Add/Update

4> **Grievance Search/Solution Ranking** – Relevance algorithm for searching and sorting of Grievances and Solutions

2.3 Data Model and Design Considerations

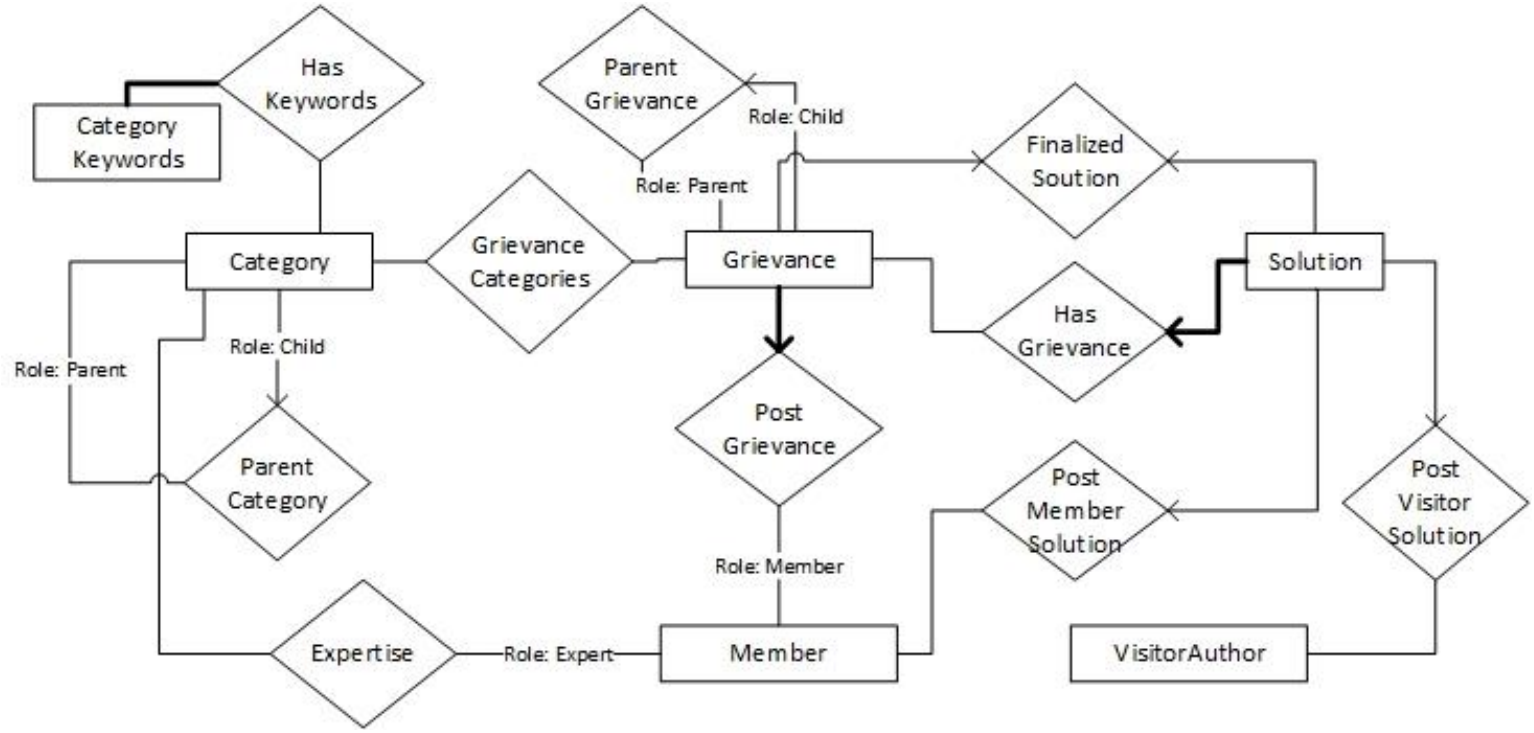


Figure 2.3.1 sortmeout ER model

2.3.1 Key Constraints

author (<u>id</u>)	identification data entered voluntarily by a visitor while posting a Solution
member (<u>id</u>)	identification data of registered users (Member/Expert)
expertise (<u>id</u>, ex_id, cat_id, FOREIGN KEY (ex_id) REFERENCES member(id), FOREIGN KEY (cat_id) REFERENCES category(id))	an Expert signs up as an expert in select categories with his (expertise) level in each; whenever a Solution is finalized fully/partially, expertise points are awarded to the Member authoring the Solution
grievance (<u>id</u>, prnt_gr_id, ath_id, fnl_sl_id, FOREIGN KEY (fnl_sl_id) REFERENCES solution (id), FOREIGN KEY (ath_id) REFERENCES member (id), FOREIGN KEY (prnt_gr_id) REFERENCES grievance (id))	Only a registered user can post a Grievance. A new Grievance posted by a Member creates a new Grievance-thread. A Grievance can have at most one finalized Solution, which either closes the Grievance-thread (awards full expertise points to the authoring Member) or prompts for an interim Grievance (awards partial expertise points to the authoring Member). An interim Grievance has one and only one parent Grievance. ($\text{level}_{\text{new grievance}}$ = 0 $\text{level}_{\text{interim grievance}}$ = $\text{level}_{\text{parent grievance}}$ + 1)
griscat (<u>id</u>, gr_id, cat_id, FOREIGN KEY (gr_id) REFERENCES grievance (id), FOREIGN KEY (cat_id) REFERENCES category(id))	A Grievance can fall into multiple Categories.
solution (<u>id</u>, gr_id, ath_id, vath_id, FOREIGN KEY (vath_id) REFERENCES author(id), FOREIGN KEY(ath_id) REFERENCES member(id), FOREIGN KEY (gr_id) REFERENCES grievance (id))	A Solution can be posted by either a registered user (authoring Member) or a visitor (authoring Author). Every Solution is associated with one and only one Grievance.
category (<u>id</u>, prnt_cat_id, FOREIGN KEY (prnt_cat_id) REFERENCES category (id))	A Category may have at most one parent Category ($\text{level}_{\text{subcategory}}$ = $\text{level}_{\text{parent category}}$ + 1)
catkeys (<u>id</u>, cat_id, FOREIGN KEY (cat_id) REFERENCES category(id))	A Category may have some pre-defined set of keywords

2.3.2 Participation Constraints

grievance (postgrievance)	Every Grievance has an authoring Member
solution (hasgrievance)	Every Solution has an associated Grievance
solution (postsolution)	Every Solution has an author (Author/Member)
catkeys (haskeywords)	Every CategoryKeyword (Catkeys) has an associated Category

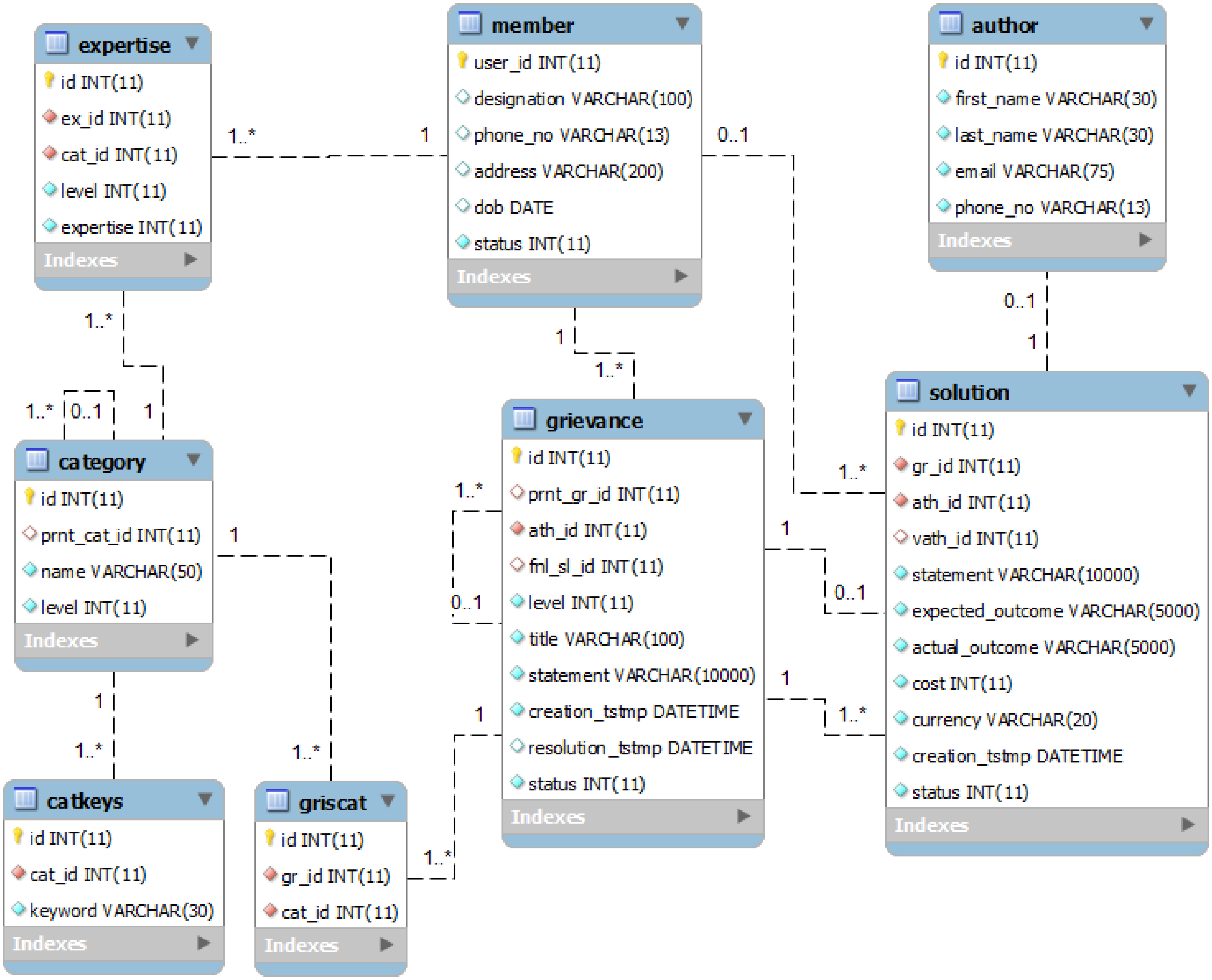


Figure 2.3.2 sortmeout relational model

2.3.3 Other Non-trivial Considerations

A new Grievance starts at level 0. Every interim Grievance is parented by a Grievance at the level immediately above it.

Categories at level 0 do not have parent Categories. Every Category is parented by a Category at the level immediately above it.

The currently chosen Solution to a Grievance appears in the finalized solution with the status flag of Grievance as *open* – indicating it is welcome to more Solutions. Once a Solution is finalized, the status of the Grievance is changed to *closed* – indicating that it will not accept any more Solutions but might lead to an interim Grievance.

The relevance algorithm invoked for presenting sorted Solutions and Grievance-search results makes use of the Members' Expertise (level and expertise), term frequencies of Catkeys and level of Categories.

The Administrator is a Django System Administrator, who uses the Django Admin Interface. The Moderator use cases including Review and Category/Keyword Management are not modeled in the data model since they can be handled within the Django Admin Interface external to the business logic of the sortmeout system.

The Forums are different views of the system's Grievances personalized for different users. A Member and Visitor can view a full list of public Grievances (called the Public Forum without search results). A user can narrow down this list with search criteria (called the Public Forum of search results). A Member can additionally view his full list of self-originated private Grievances (called his Private Forum; including the search tool that works on his self-originated Grievances). An Expert can have a Private Forum, can access the Public Forums and has a view of Grievances that fall within the Categories (totaling the Domain) of his Expertise.

3 MAIN SYSTEM FUNCTIONALITIES

3.1 VISITOR SCENARIOS

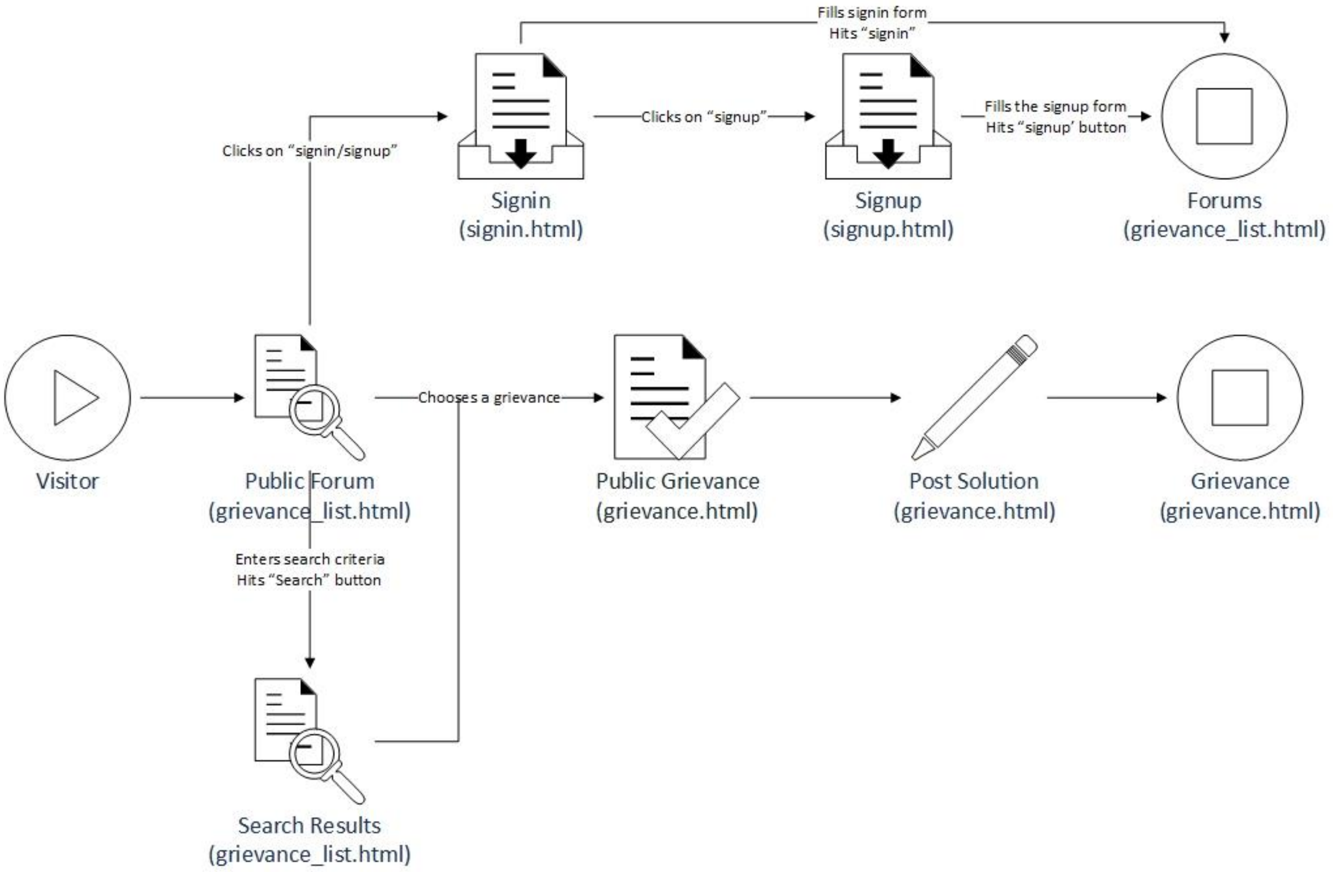


Figure 3.1 page transitions for Visitor

A Visitor to sortmeout first finds himself at the Public Forum. He may choose to search for a Grievance using the search tool. From the resulting grievance list generated by the system he may choose to view an expanded Grievance thread, which contains a flow of Grievance and interim Grievances together with their finalized Solutions at each level. If there is an outstanding Grievance he may post a Solution to the same. This Solution will be associated with the outstanding Grievance.

Alternatively the Visitor can choose to signup with/ signin into the sortmeout system. He may signup as an Expert or as a normal Member. There are different registration forms for the different types of Members. An Expert registration form is posted to the system's Moderators for review and approval.

3.2 Grievance Search

When a user enters category based search criteria, the system looks up the relevant terms from the search string into the predefined set of category keywords. Keywords of the most specific category(ies) carry the highest weightage, which trails off with Category level. Miscellaneous terms could be ranked using tf/idf (term frequency/inverted document frequency logic) by making use of the database tool's full text indexing on entered Grievances (now available in MySQL's InnoDB).

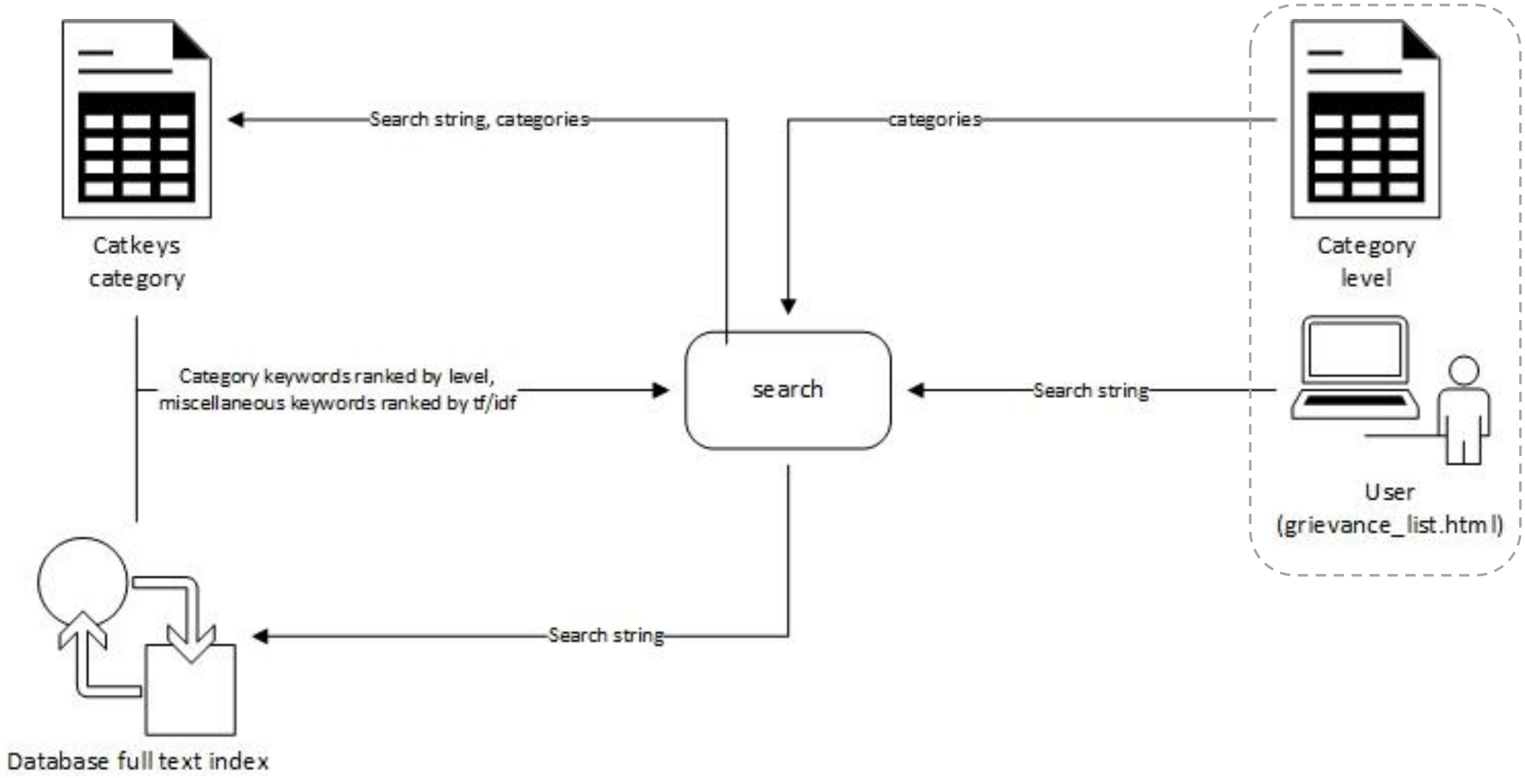


Figure 3.2 search tool

3.3 Grievance Solving (Walk-Through) Process

Once a Visitor signs in as a Member, sortmeout presents him with his Private Forum. He can choose any of his self-originated Grievances to expand its thread. If Solutions have been posted to any of his outstanding Grievances, he may choose and try or request the execution (offline) of any of them, one at a time, until he encounters one of the Solutions that solves his Grievance. He may then continue the thread by writing a next level interim Grievance or may close the thread by finalizing a Solution – both the actions lead to the deletion of the data pertaining to all the other viable Solutions at that level.

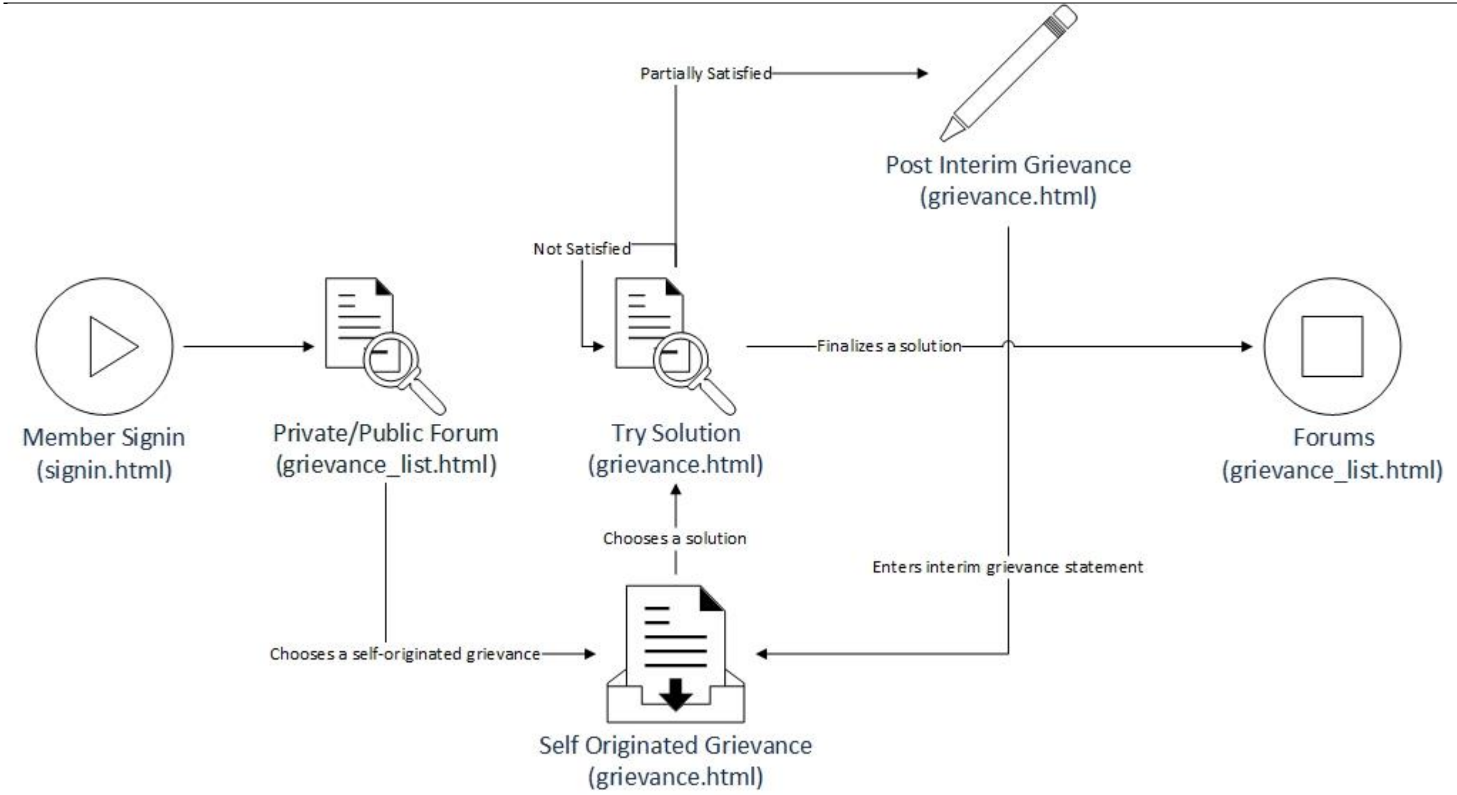


Figure 3.3 Grievance walk-through to resolution

3.4 Solution Ranking and Sorting

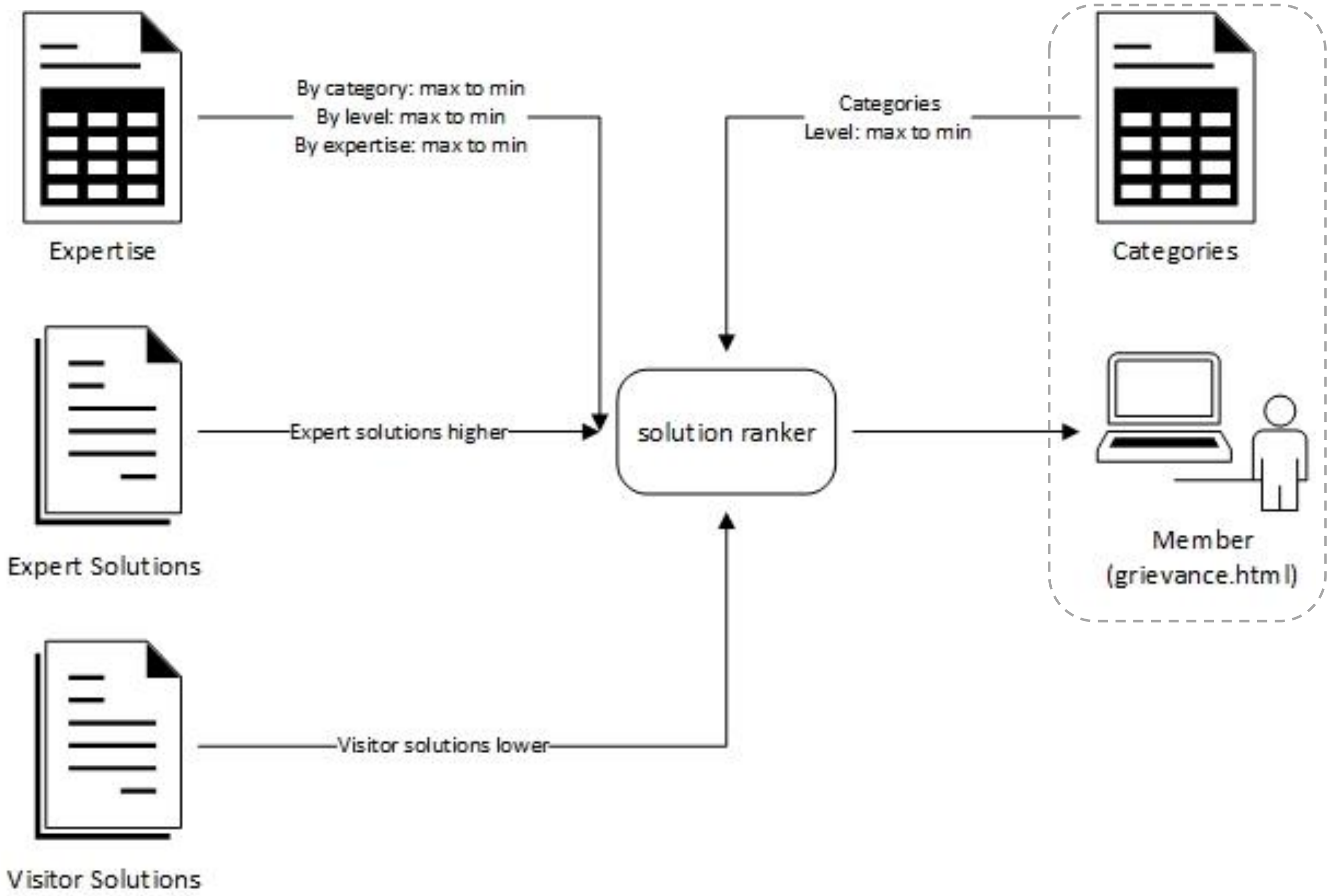


Figure 3.4 sorting of Solutions

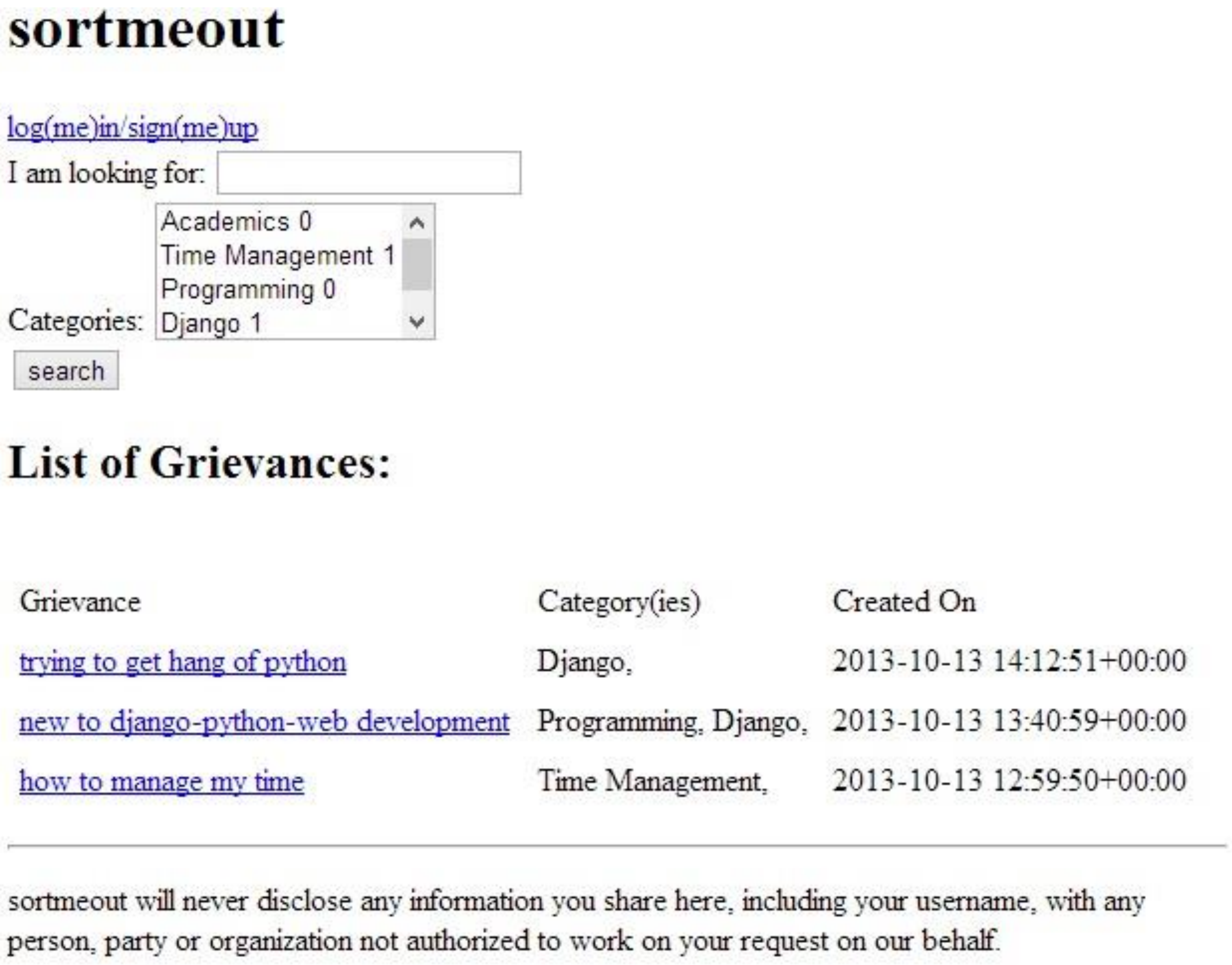
Whenever a Member expands his Grievance thread he would want any outstanding Solutions to be ranked for his comfort. The Solution Ranking module fetches the Category(ies) of the Grievance thread and identifies Solutions from Experts who have the highest Expertise (level + expertise) in the most specific Category(ies) and trailing off with Grievance level, and Expertise level and finally expertise. An unsorted list of Solutions from Visitors (i.e. simply Authors) may follow the list of Expert Solutions.

4 INTERFACE

This section illustrates the chief interfaces of sortmeout and how it makes use of Django's Model-View-Controller pattern to generate webpage-content dynamically.

4.1 PUBLIC FORUM (DEFAULT/ENTRY VIEW)

The prototype webpage below illustrates the chief components of sortmeout's dashboard. It shows the default Public Forum of open grievances with a search form. From here Visitors can choose to navigate to the signup page if they want to register as Members/Experts at sortmeout; or to the signin page in order to establish themselves as Registered Users.



sortmeout

log(me)in/sign(me)up

I am looking for:

Categories:
- Academics 0
- Time Management 1
- Programming 0
- Django 1

List of Grievances:

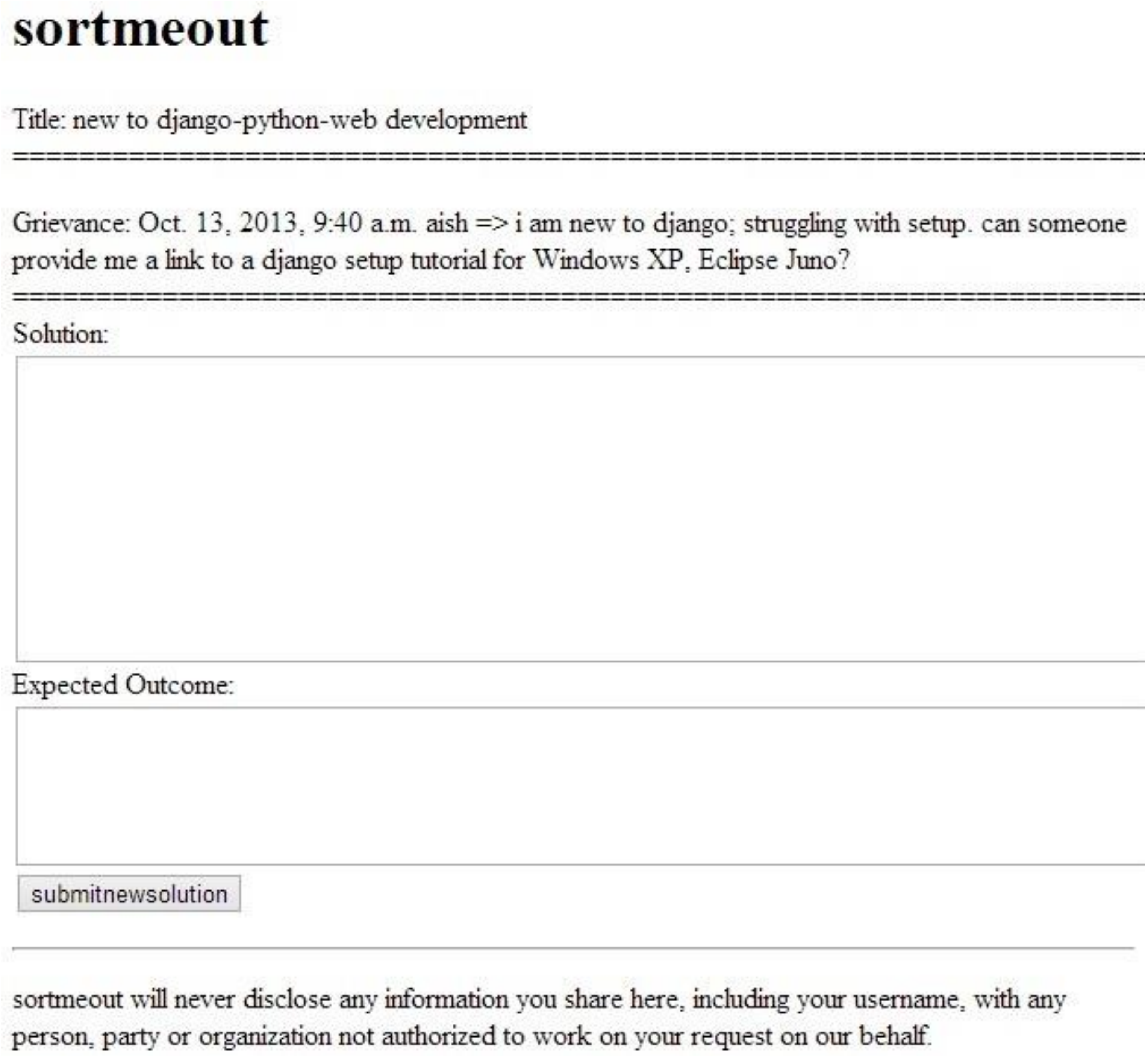
Grievance	Category(ies)	Created On
trying to get hang of python	Django,	2013-10-13 14:12:51+00:00
new to django-python-web development	Programming, Django,	2013-10-13 13:40:59+00:00
how to manage my time	Time Management,	2013-10-13 12:59:50+00:00

sortmeout will never disclose any information you share here, including your username, with any person, party or organization not authorized to work on your request on our behalf.

Figure 4.1 grievance_list.html (the entire Public Forum shown as the index.html of sortmeout)

4.2 Visitor view of a Grievance

If a Visitor chooses to expand a Public Grievance from the dashboard, the system will generate a customized view of the Grievance thread suitable for public viewing (identity of the originating Member may be hidden for the sake of privacy). If the Grievance is open, the grievance.html will contain a form for entering a new Solution to the Grievance. This view is similar to the one offered to an Expert except that it may contain the originating Member's identification data.



sortmeout

Title: new to django-python-web development

Grievance: Oct. 13, 2013, 9:40 a.m. aish => i am new to django; struggling with setup. can someone provide me a link to a django setup tutorial for Windows XP, Eclipse Juno?

Solution:

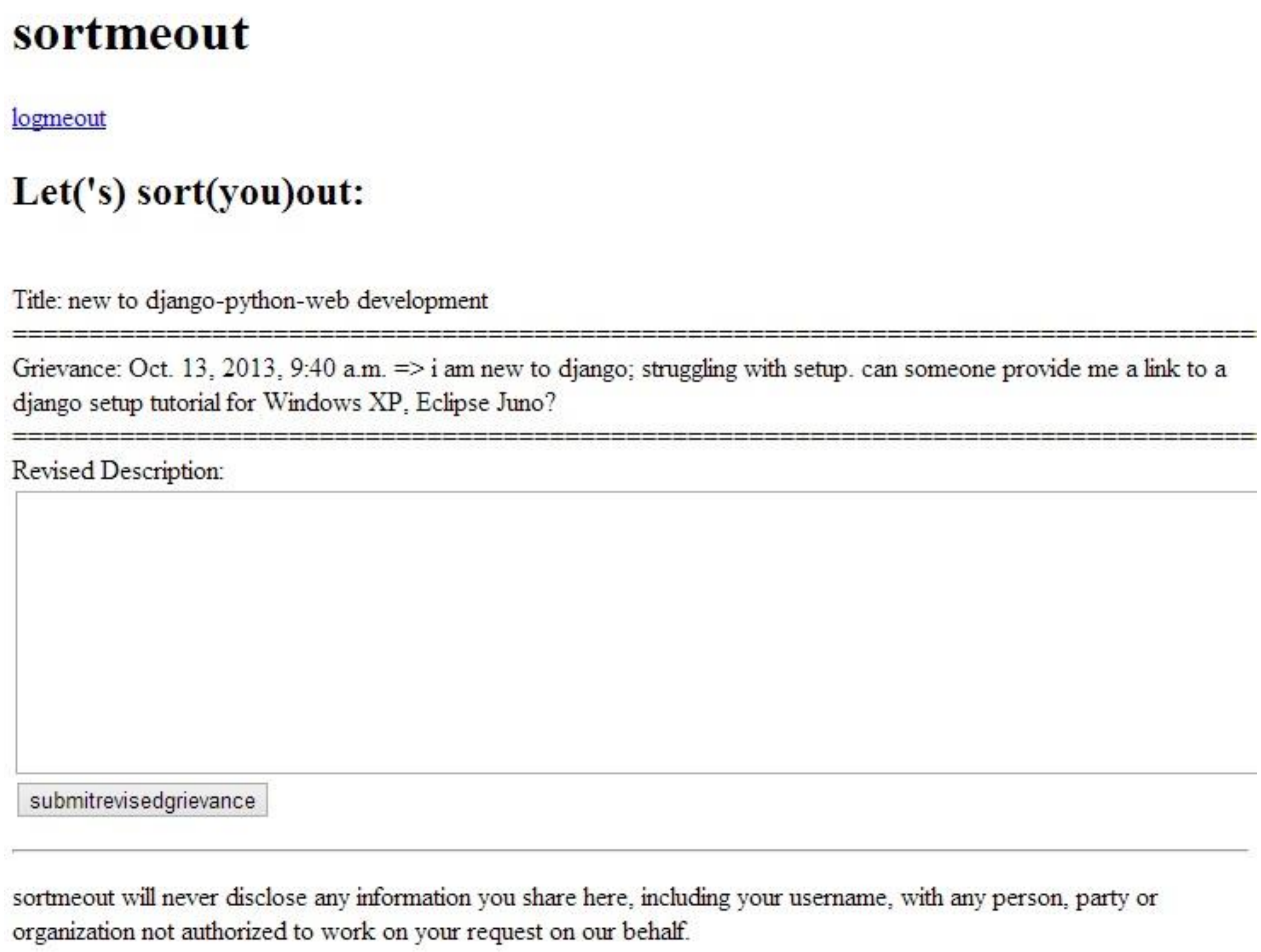
Expected Outcome:

sortmeout will never disclose any information you share here, including your username, with any person, party or organization not authorized to work on your request on our behalf.

Figure 4.2 grievance.html (when expanded by a Visitor/Expert)

4.3 Grievance-originator Member's view of a Grievance

When a Grievance thread is expanded by the originating Member himself, the system generates another view of the Grievance thread fit for the Member's walk-through. It will allow the Member to choose Solutions for the outstanding Grievance and/or provide a description of an interim Grievance.



sortmeout

logmeout

Let('s) sort(you)out:

Title: new to django-python-web development

==

Grievance: Oct. 13, 2013, 9:40 a.m. => i am new to django; struggling with setup. can someone provide me a link to a django setup tutorial for Windows XP, Eclipse Juno?

==

Revised Description:

sortmeout will never disclose any information you share here, including your username, with any person, party or organization not authorized to work on your request on our behalf.

Figure 4.3 grievance.html (same Grievance when expanded by the originating Member)

5 Required Technologies

1> **Django** 1.5.4 – Business Logic; Data Model; View-Controller
2> **Python** 2.7.3 – Business Logic; Webpage-Server-Scripting
3> **MySQL** (InnoDB) – Database Server
4> **XHTML**, **CSS** – Webpages (Views)