

User Stories:

AS A LOGGED IN USER:

- I want my tasks to be auto-categorized, so I don't have to think about it.
  - I want to **see** all my tasks, because I want to keep track of my tasks.
  - I want to **add** a task, because I've got a new task.
  - I want to sort my tasks, because I want to focus on one category at a time.
  - \* I want to prioritize certain tasks, because I want to know what to focus on first.
  - I want to **update** my tasks, because I may have made a mistake or changed my mind.
  - I want to change my category, because it has been mis-categorized.
  - \* I want to change my title, because it has been mis-labeled.
  - I want to **delete** my tasks, because I want to plan ahead.
  - I want to be able to mark my tasks as complete, because I want to keep track of my productivity.
  - \* I want to see my completed task, because I want to see what I've accomplished.
- 
- I shouldn't be able to edit other users' posts, because I don't own those posts.

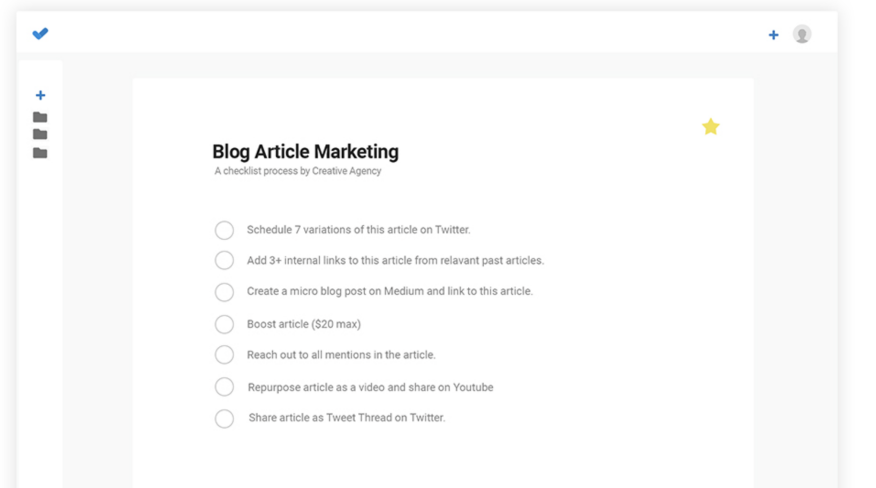
ERD:


User	
PK	<u>UniqueID</u>
	name
	email
	password

Task	
PK	<u>UniqueID</u>
FK	user_id
	category_name
	title
	note
	is_completed
	is_deleted
	priority
	date

Priority either boolean or 1 - 3 (default 1)

Browse	GET	/tasks
Read	GET	/tasks/category
Edit	POST	/tasks/:id
Add	POST	/tasks
Delete	POST	/tasks/delete?





login/logout

HOME/CATEGORY NAME

new task!

home

category  
button  
(icon)

category

category

category

completed

task title

category (drop-down?)

date

priority?

task 1

☐

task 2

☐

task 3

☐


task name

category

date

☆

edit



(slide out form)

footer (links) (our names)

## Option 6: Smart TODO List

When you are recommended something it's not always easy to jot it down for later in an organized fashion. Adding the item to your phone or computer ends up taking time and opening up the right app is only part of the problem. You then have to locate the right list ("Movies to watch", "Books to read", etc.) to add to. And if you do get it in to the right list, you don't have much more context about it. This delay and lack of additional information acts as a huge deterrent.

The solution? A smart, auto-categorizing todo list app. The user simply has to add the name of the thing, and it gets put into the correct list.

### Requirements:

Each todo created should be categorized as one of:

1. Film / Series (To watch)
2. Restaurants, cafes, etc. (To eat)
3. Books (To read)
4. Products (To buy)

In order to determine the category the app will probably need to use various API services such as those offered by Google, Wolfram Alpha, Rotten Tomatoes, Amazon, Yelp and others.

API services mentioned above are only suggestions. You will have to investigate how to balance the accurate categorization of items with having to deal with multiple API endpoints.

Users should be able to change a category of an item in case it was mis-categorized or could not be categorized at all.

~~Users should be able to register, log in, log out and update their profile.~~

<https://products.wolframalpha.com/api/documentation/>

[https://api.wikimedia.org/wiki/API\\_reference/Core/Pages/Get\\_description](https://api.wikimedia.org/wiki/API_reference/Core/Pages/Get_description)

<https://console.cloud.google.com/apis/library/customsearch.googleapis.com?supportedpurview=project>

New Task	Existing Tasks
<p>HTML Form:</p> <p>Text input (action:POST)</p> <p>Category dropdown: selection</p> <p>Priority dropdown:</p> <p>Submit button</p>	<p>Route (get req) to ('/') calls helper function passes TemplateVars to EJS file</p> <p>New files in Routes folder</p> <p>Require path in Server.js follow examples (require, app.use)</p> <p>Require in helper function (below) from queries folder</p>
<p>Express post route</p> <p>Loads page of associated category or home (if none)</p>	<p>JS/SQL Func queries DB and gets the obj return from DB</p> <p>New file under Queries folder</p>
<p>JS helper funcs to check if there are keywords in title</p>	<p>Html/EJS loops through each task</p>
<p>SQL Add task and associated data to db</p>	<p>CSS formatting</p>
<p>API calls to check if title matches category</p>	
<p>CSS formatting</p>	
<p>Drop down functionality (jquery?)</p>	

Sooyeon

Rowan

## Intro:

Introduce ourselves

*This is our app **smart to-do list**, a task list that auto-categorized your tasks to make it easier to keep track of what you need to get done.*

*We chose the to-do list because we wanted to challenge ourselves, something outside of our comfort zone.*

*This includes working with APIs and building a single page web app.*

*I focussed on building out the API functionality as well as formatting the design based on our initial wireframe*

*Rowan focussed on planning out what features we worked on each day and implementing the SQL, and asynchronous functions*

*We collaborated on the filtering and implementation of the backend*

## Demo:

Load in sample tasks (reset db)

### Section 1: Adding tasks:

#### First task:

- Explain title, category options, priority

- Default category is uncategorized, auto-categorizes when you click "Submit"

- Priority determines sort order

- Show editing/deleting tasks

To show off the API functionality, use these tasks:

Eg:

The Sopranos (movies/tv)

Order Dominos (restaurants)

Read The Hobbit (books)

### Section 2 categorization, completion and editing:

#### Categorization pages:

- Go through each category page

- Complete a task on the home page

- Show Completed page

- Show completed page and that it shows there

- Uncomplete a task, redirecting you to the home page

Conclusion:

We focussed on simplicity and user experience

Keeping the design lightweight for ease of use

Some potential stretch functionality would be:

Adding user login

Custom categories

Ability to add notes

API loads associated information with the tasks into notes (eg. movie/book information)

— What you learned:

Implementing API, getting it working asynchronously. Single page app functionality.

Hope you liked it!