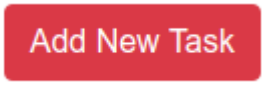
Architecture Design

How does your system work?

The application is built on a MVC architecture. There is a Model,Controller, and View(In Client Folder) called to do list. The client applications acceses a REST API in order to Create, Read all (To populate the list table), Update and Delete list items. A table listing the routes can be found below or you can view the routes in routes.js.

In structions on how the rest api interacts with user actions is given below.

CRUD	Route	method	How to trigger events
Create	/task/add	POST	 Fired when user clicks on above icon.
Read	/tasks	GET	Fired when page loads
Update	/task/edit:id	POST	 Fired when user clicks on above icon.
Delete	/task/delete:id	DELETE	 Fired when user clicks on above icon.



** When the add task button is clicked a new*  *record is crated inside the html table. The record isn't to the database until the user clicks .*

What considerations you need?

Keep in mind this is a very basic application thats built on top MVC framework utilizing technologies such as NodeJs,MySQL,Jquery,CSS,HTML and Boostrap. It uses webpack to create a bundle file that is used by the client browser.

In order to work on the application you need an understanding of nodejs, bootstrap,jquery, express, ejs(Templating engine), css, html, mysql, sequelize.

Limiations of the Application?

It is a very simple application that uses one table. It has no error checking capabilities, security features, or a relational design. The application can only ever have one ToDo List. Users can't create multiple ToDoList.

As it is now, the application is running on Heroku using a free account which is a bit slow I think.

If you wanted to create the application for multiple users you would need to redesign the database, add a login feature, and create user table, todolist_table, task table(Foreign Key to ToDoList table).