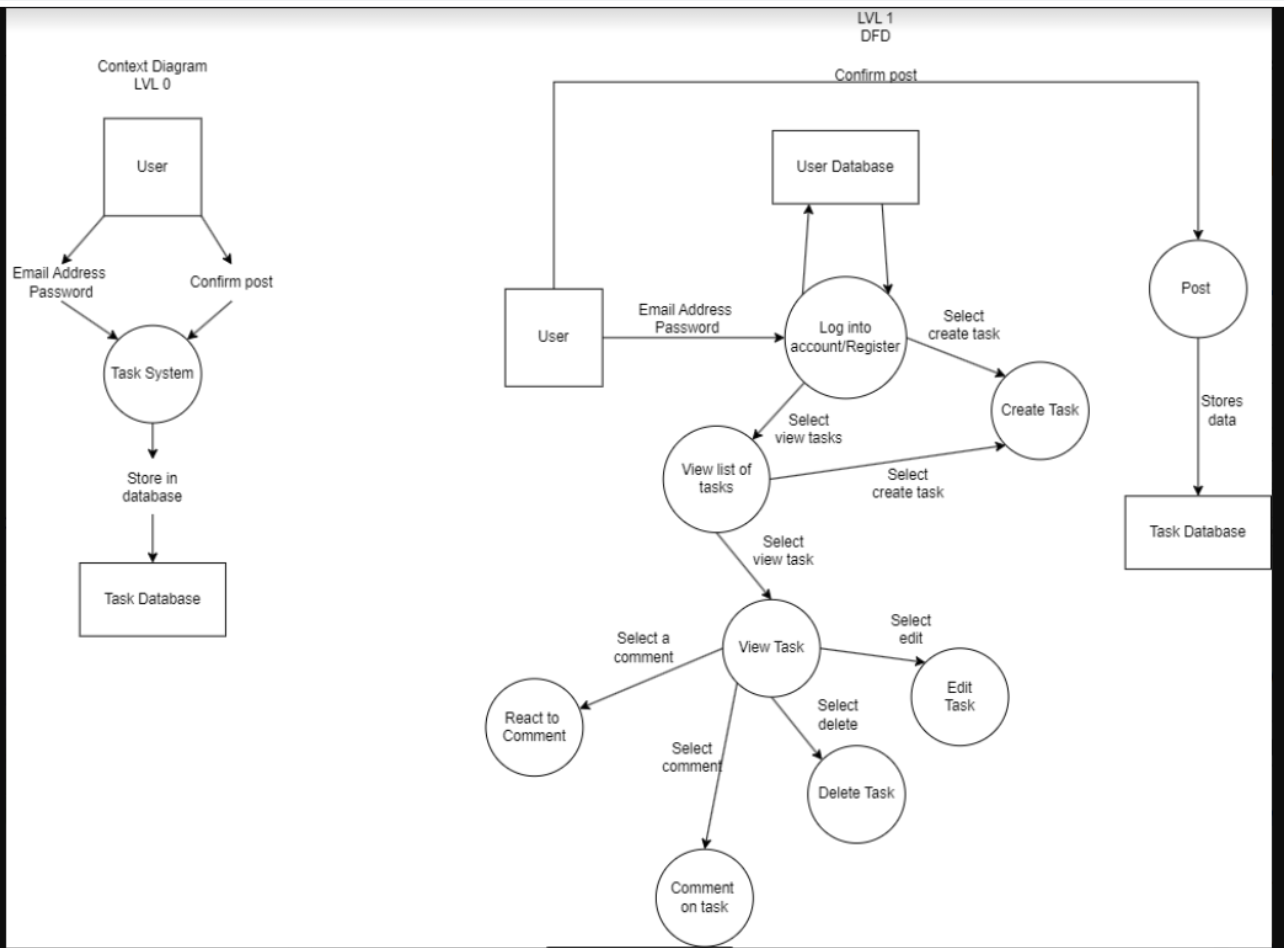
# Catch-22

[Andrew Schulz](mailto:aschulz6@uncc.edu), [Lutou Xiong](mailto:lxiong13@uncc.edu), [Irzem Ahmed](mailto:mahmed29@uncc.edu), Kevin Salley

### **Data Flow Diagram**

(2 points) Create a context diagram

(2 points) level 1 diagram showing the flow of information and processes of the system.



### **Models and Associations**

(2 points) Define the models you will need in your application. Think of the different things you want to represent in your system and what properties they have. Be sure to include all model attributes and model attribute data types.

* Home page
* Introduction to the website
* Link to homepage,user projects,accounts
* Edit/new page
* User enter data for task/projects
* Has save and back buttons
* Update page
* User can enter data/information
* Has a update and back button
* Comment page
* User can comment on task
* Has a share and back button
* Can view the task
* View list of project page
* User can view the project
* Has 4 sections backlog, todo, in progress, and done
* Under each section user are allowed to edit or comment
* View project/task page
* User can view descriptions of the task/project
* Has a back button
* Login in page
* User can log in
* Sign in and register button
* Registration page
* User can create an account
* A register button

(1 Point) Decide which associations between the models are appropriate using the Basic Relationship Pattern page.

* Many to one

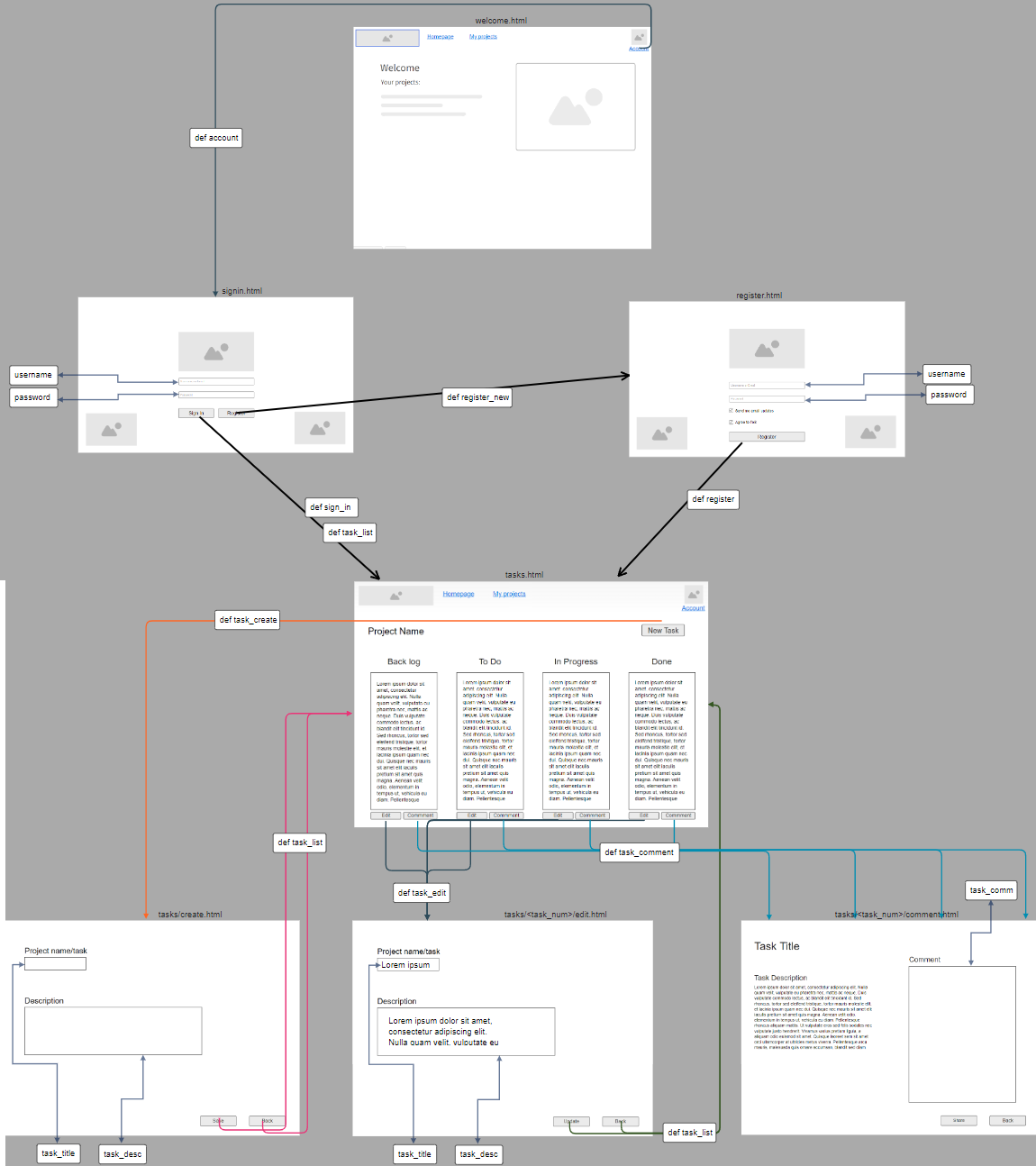
### **Update Lo-Fi UI prototypes**

1 - (1 point) function name for any UIs that will correspond to a route.

2 - (1 point) View name -- e.g., index.html

3 - (1 points) Database field names for all input fields. These would represent the properties/fields/attributes of your models.

4 - (1 points) route action (method) for any buttons or links that correspond with actions in the app. i.e. A "show" link may correspond with the show method in a particular route. Remember your CRUD actions!



### 

### 

### 

### 

### **Update Backlog**

