

# What to Do?

---

## Section 1: Functionalities

### Data

- Data is saved locally
- Data is not cloud supported

### Account Verification

- No verification/account hashing
  - future work would include implementation with a cloud database. Ideally with fireBase.

### Functions

- Users can create tasks (generate cells in UITableView object)
- Users can add details to tasks using keyboard
- Users can edit details to tasks using keyboard
- Users can not remove tasks (delete cells in UITableView object)
  - ~issue with enumerations problems when indexing UITableView and keys
- There is a login screen
  - ~No actual functionality and demonstrates basic navigation
- There is an account creation screen
- There is a main/home screen
  - ~Future iterations would improve UI elements
- Basic background image
  - ~issues with applying to login and creation screens

## Section 2: Screen Archetypes & Design

### Login

- Users can login..
- ..or create an account..
- ..or skip the process altogether

### Create Screen

- User account creation
- Assign userName to accountID

- Or skip the process

### Main Screen

- Data (lists) is requested from local or cloud database and displayed using a table view/list view for the user
- Lists can be created, edited, or deleted

### List Screen

- Displays a detailed view of a list upon user selection.
- Users will have the option to add, edit, or remove entries in the list.

### Details

- Displays a detailed view of an entry of a list upon user selection

### Navigation

Login -> Account Creation or Main Screen

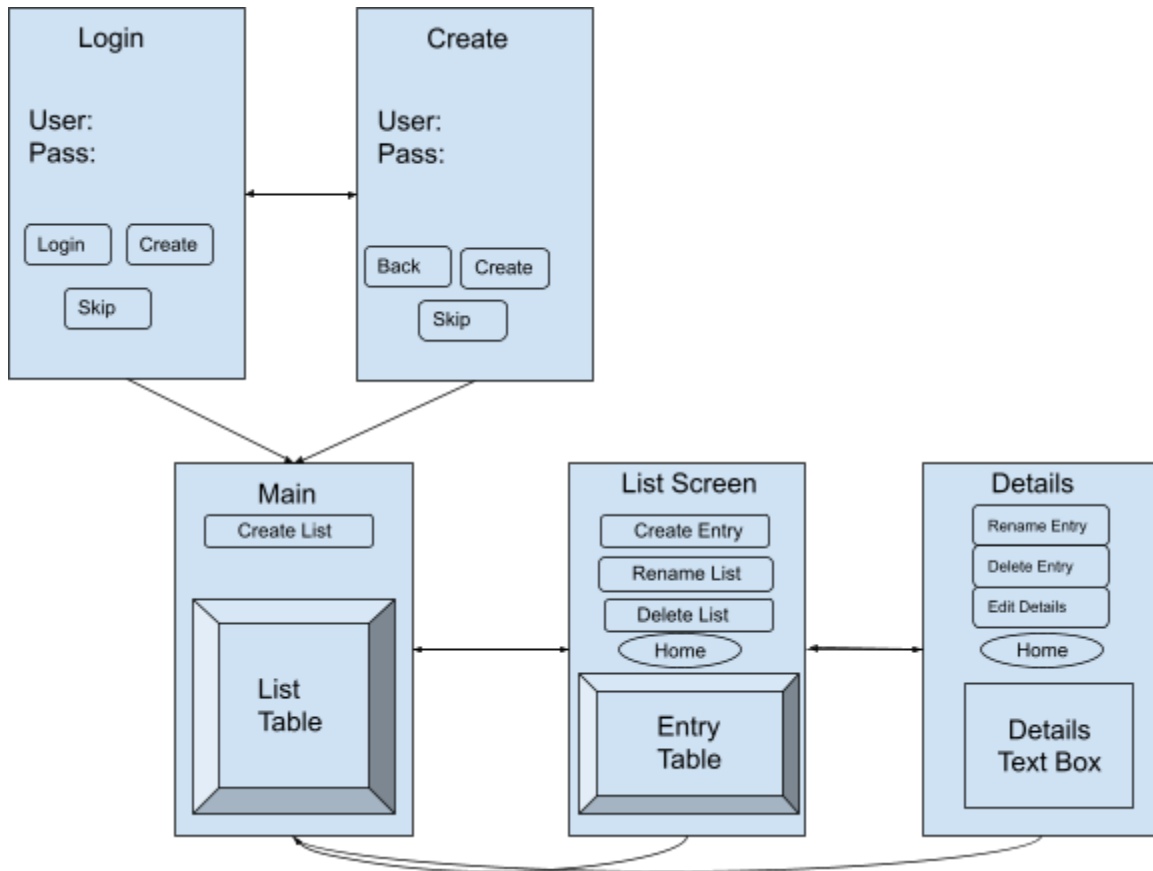
Account Creation -> Main Screen

Main Screen -> List Screen -> Details

## Wireframe/Schema

---

| Property    | Type    | Description   |
|-------------|---------|---|
| listName    | String  | Name and acting ID for a user's list                              |
| userName    | String  | The name for login and dictionary key for corresponding accountID |
| userPass    | String  | Alpha-numerical password for account verification                 |
| entryName   | String  | The title for each entry in a given list                          |
| entryDetail | String  | Text for the details of a given entry                             |
| accountID   | Integer | Uniquely generated number for each user                           |



## Section 3: Github Location

<https://github.com/duck-mon/whatToDo>

## Section 4: Test Results

### → Scenario 1: Login screen

- ◆ Data: username, password
- ◆ Pass?: Failed, functionality does not work
  - Caveat, button does progress to the main screen

### → Scenario 2: Account Creation

- ◆ Data: username, password
- ◆ Pass?: Failed, functionality does not work
  - Caveat, button does progress to the main screen

### → Scenario 3: Skip Login (Guest User)

- ◆ Data:
- ◆ Pass?: Pass, functionality does work. “Skip” button does proceed to the main screen

#### → Scenario 4: Task creation

- ◆ Data: tableView: UITableView, userDefaults
- ◆ Pass?: True, functionality does work. EntryViewController is instantiated and user input is saved to the tableView in the main screen. Data saved using userDefaults

#### → Scenario 5: Detail Creation

- ◆ Data: details: UITextView, tableView: UITableView
- ◆ Pass?: True, functionality does work. Users can click/select a given cell to enter a TaskViewController that has a textView that is selectable and editable

#### → Scenario 6: Detail Editing

- ◆ Data: details: UITextView, tableView: UITableView
- ◆ Pass?: True, functionality does work. Users can save data written on the details textView element by hitting the "Save & Quit" button on the top right corner of the view.

#### → Scenario 7: Persistent Data

- ◆ Data: details: UITextView, tableView: UITableView
- ◆ Pass?: True, functionality does work. Closing and relaunching the application shows that data is persistent to local data.

## Section 5: Future Work

- Firebase implementation
- enhanced UI/UX
- Exportation of lists
- Implement account hashing locally then with Firebase
- removal of tasks