Project Development Sheet

“server-side” git front-end branch

---

environments:

- node.js
- express
- mongoose

---

entities:

```
List {
        id_list: int,
        name_list: string,
        items: Items[ ]
}

Items [ ] {
                id_item: int,
                name_item: string
}
```

---

endpoints_routes:

- Fetch all lists (**GET**)
  —> url: ‘/lists’, method: findAllLists()
- Fetch list by id (**GET**)
  —> url: ‘/lists/:id_list’, method: findListById(id_list)
- Add new list (**POST**)
  —> url: ‘/lists/’, method: addList(list_name)
- Edit list name (**PUT**)
  —> url: ‘/lists/:id_list’ , method: updateListName(id_list, name_list)
- Delete list (**DELETE**)
  —> url: ’/lists/:id_list’, method: deleteList(id_list)

- Fetch all items (**GET**)
  —> url: ‘/items’, method: findAllItems()
- Fetch item by id (**GET**)
  —> url: ‘/items/:id_item’, method: findItemById(id_item)
- Add new item (**POST**)
  —> url: ‘/items/’, method: addItem(id_list, item_data)
- Edit item in list (**PUT**)
  —> url: ‘/items/’, method: editItem(id_list, id_item, name_item)
- Delete item in list (**DELETE**)
  —> url: ‘/items/’, method: deleteItem(id_list, id_item)

“client-side” git back-end branch

environment:

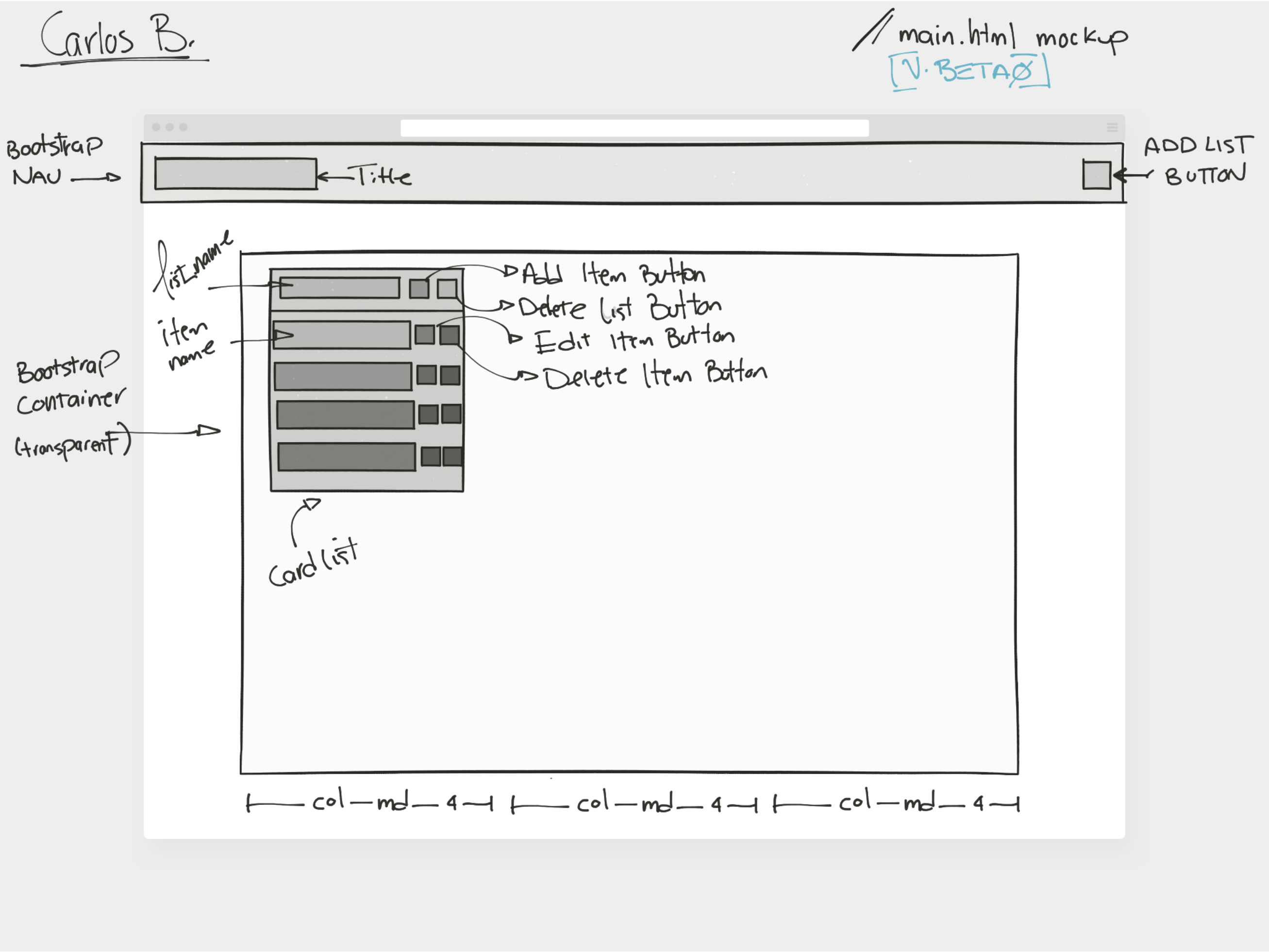
- Angular JS or Vanilla JS

views:

- Show all lists with elements.
- Add new list button.
- Edit list button.
- Inside list item, add, edit, delete item buttons.
- Modal for UI interaction.

mockup:

./main.html


Carlos B.
// main.html mockup
[V.BETAØ]
Bootstrap NAV
Title
ADD LIST BUTTON
list_name
item name
Add Item Button
Delete List Button
Edit Item Button
Delete Item Button
Bootstrap Container (transparent)
Card list
col—md—4
col—md—4
col—md—4