**Team NaCl**

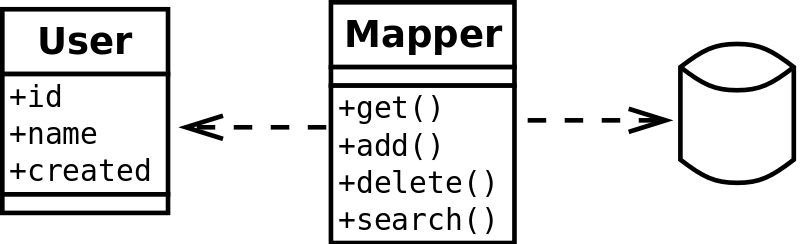
Software Detailed Design

Group Leader: Martin Bacala

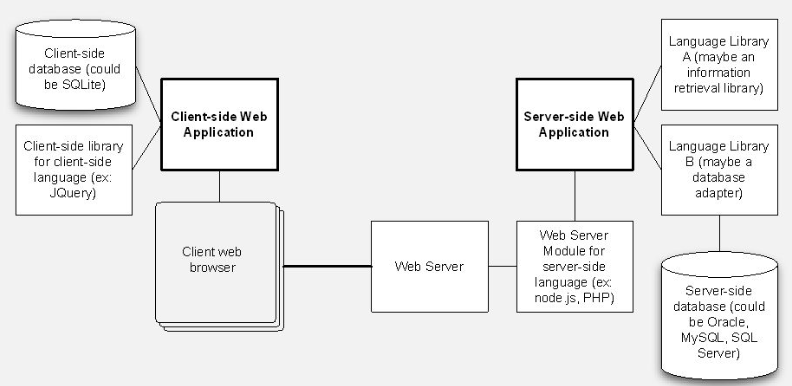
Members: Nick Manolov, Hung Nguyen

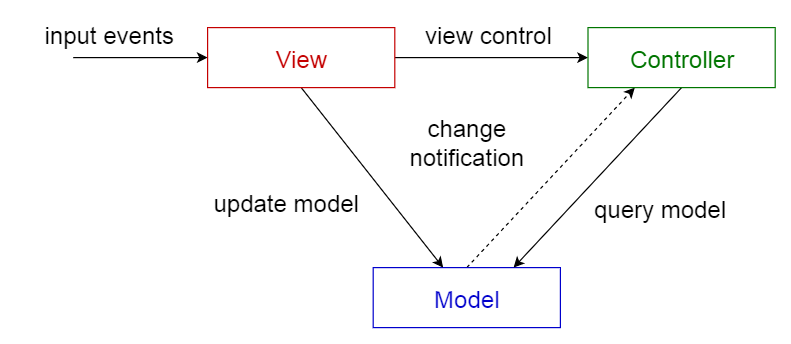
**Data Design**

Our application will use the Data Mapper architectural pattern. The interface would include functions such as Create, Read, Update, and Delete, that operates on objects that represents entity types within the data store.



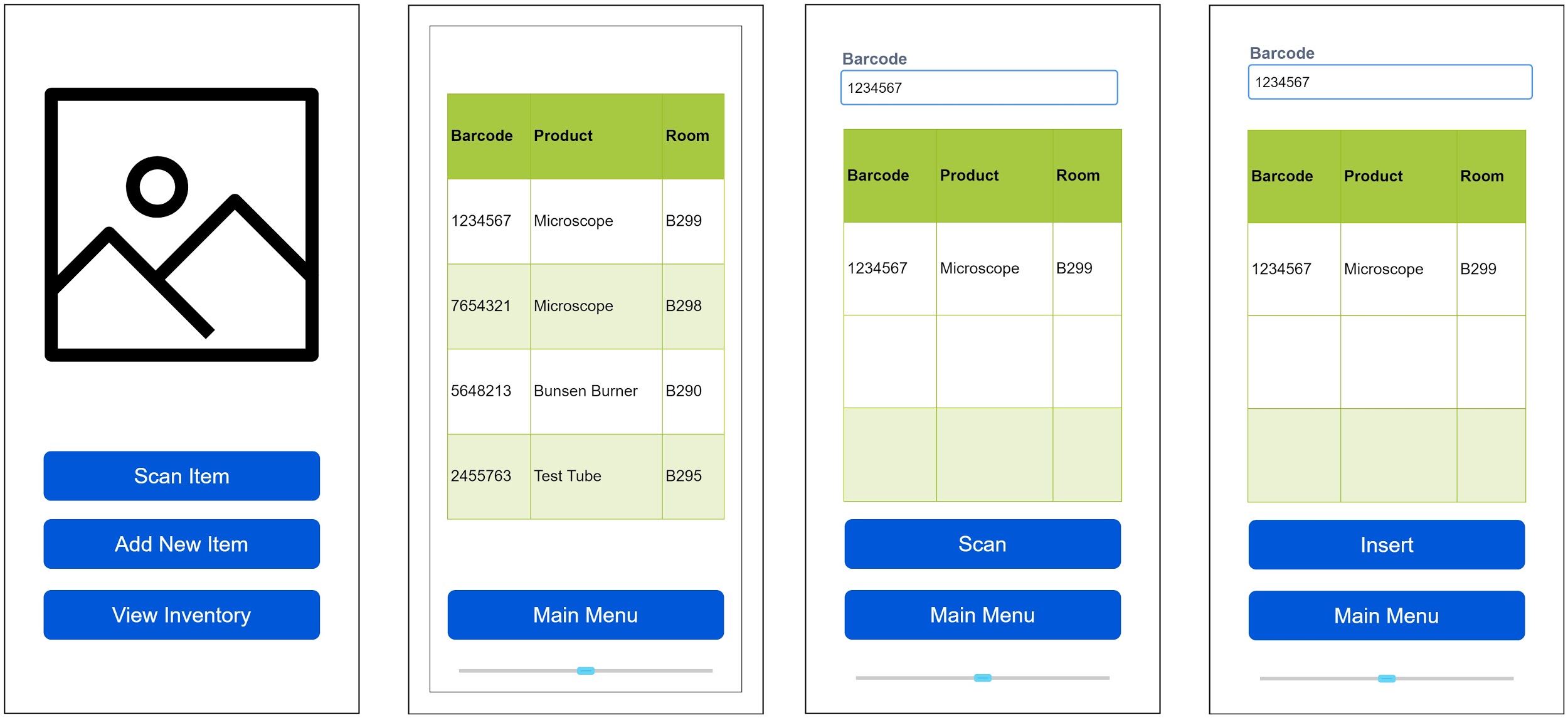
**Architecture Pattern**

****



**Interface Design**

Our app will contain several different user interfaces that will aid with viewing the inventory, adding to the inventory and editing the inventory. Each one of these pieces of functionality will have their own interface. The diagram below shows the early version of the interfaces we plan on implementing.



**Procedural Design**

In order to design programmatically or procedural the user interface you must understand the framework and to know the relation between user interface elements and their corresponding classes. Also the parent-child relation between elements is important, as some controls have containers or collections used to manage other child controls.

We plan on leveraging the Object Relational Mapper. By using this design pattern, we will query virtual objects from a virtual object database instead of explicitly calling the database itself. This will add a layer of abstraction between the user and the database which protects the database from SQL injection attacks.