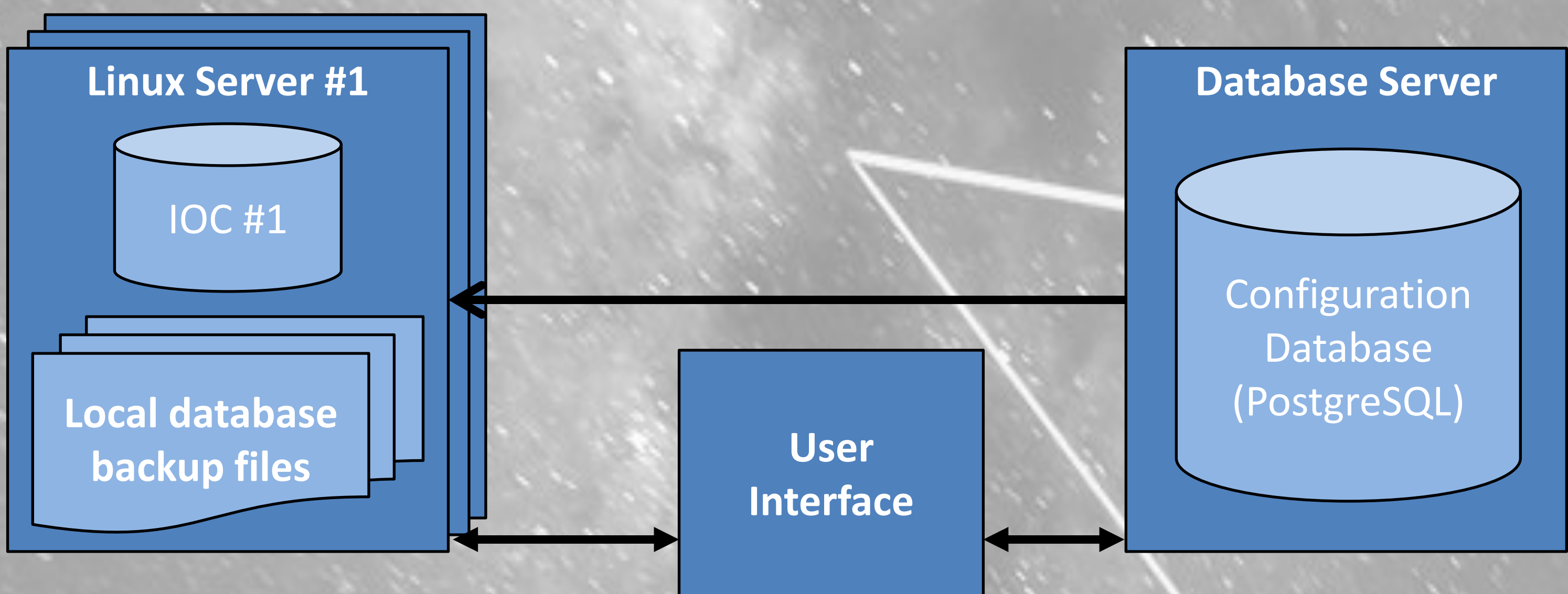


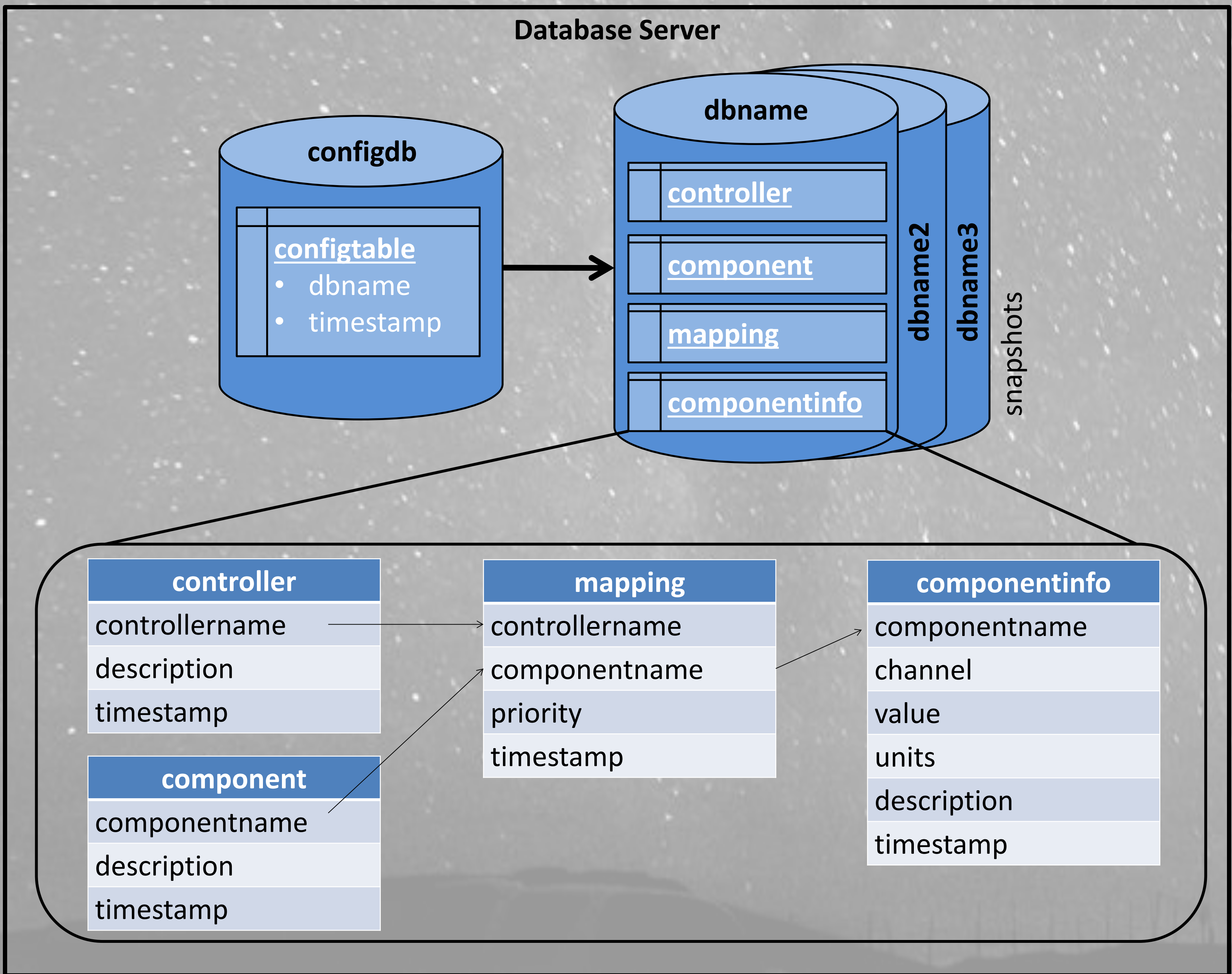
Overview

- Manages persistent configuration data through a database-backed solution with a simple API and user interface
- Supports fail-safe operations when the database server is down or there is a network glitch



Database

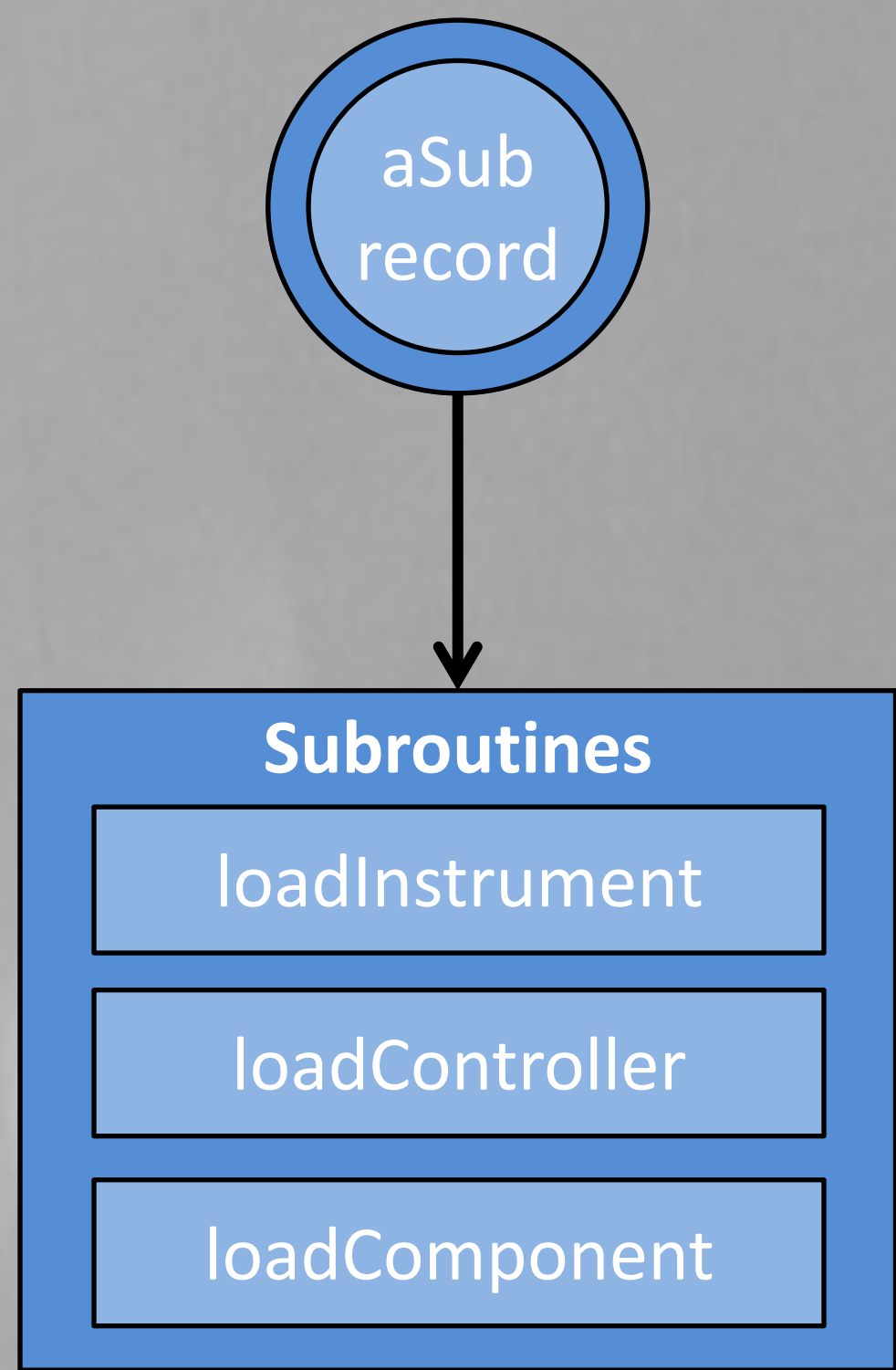
- Allows easy hierarchical configuration with a controller/component relational schema
- Allows prioritization of component load order
- Uses triggers to provides change history, time-stamping and “house cleaning”
- Ability to create and load snapshots



IOC

- Configuration support calls can be initiated from within the iocsh or dynamically using EPICS record support

```
k0:ioc> loadController(k0, supHIRES)
csTool: Connecting to configdata database
csTool: 1 components returned for supHIRES
csTool: Component = tcsHIRES
csTool: 39 channels loaded for tcsHIRES
k0:ioc>
```



User Interface

- Written in C++ using the Qt framework
- System level view
- IOC comparison with unit conversion
- Ability to add, update and delete items

