DESIGN DOCUMENT

**Contents**

[Power Grid Monitor and Simulator Requirements 2](#_Toc483499127)

[**Entities Required** 2](#_Toc483499128)

[**Grid Monitor** 2](#_Toc483499129)

[**Grid Controller** 3](#_Toc483499130)

[**Consumer Simulator** 4](#_Toc483499131)

[**Data Writer** 4](#_Toc483499132)

[**Incident Logging** 4](#_Toc483499133)

[**CRUD Features and/or Application Builder** 5](#_Toc483499134)

[**Production and Testing Modes** 5](#_Toc483499135)

[**Others** 5](#_Toc483499136)

[**Putting It All Together** 5](#_Toc483499137)

[Grid Model Diagram 6](#_Toc483499138)

[Power Grid GUI Mockup 7](#_Toc483499139)

[Power Grid CRUD Mockup 8](#_Toc483499140)

[**Home** 8](#_Toc483499141)

[**Plant** 9](#_Toc483499143)

[**Substation** 10](#_Toc483499145)

[**Feeder** 11](#_Toc483499147)

[**Customer** 12](#_Toc483499149)

# **Power Grid Monitor and Simulator Requirements**

## **Entities Required**

**RQ-01**

* The application should have entity classes to represent the components of the grid.

1. Grid - ID, name, description.
2. Plant - ID, status, fuel type (energy source), power production rating, efficiency rating, maximum capacity (speed), current speed, current power output, fuel level, fuel consumption rate, latitude/longitude, and parent Grid.
3. Substation - ID, status, current load (demand) of all consumers attached to the substation, maximum load, latitude/longitude, and parent Grid.
4. Feeder Line - ID, status, current load (demand) of all consumers attached to the line, maximum load, and parent Substation.
5. Consumer - ID, status, consumer type, current load (demand), maximum load, latitude/longitude, parent Feeder.

## **Grid Monitor**

**RQ-02**

* The monitor should check for component heartbeats frequently by reading the "heartbeat" field in the entity objects. The heartbeat frequency should be specified in the monitor's configuration file. Let's call this frequency ***H*** (for Heartbeat).

**RQ-03**

* Less frequently than heartbeats, the monitor should read production and consumption data for the relevant components (from the relevant fields in the entity objects). Again, the frequency should be specified in the configuration file. Let's call this frequency ***D*** (for Data).

**RQ-04**

* In addition to the production and consumption data on the components, there should be a prominently-displayed summary section that shows: the total energy produced, the total energy consumed, and the variance (difference). A properly operating grid has a variance close to 0 at all times.

**RQ-05**

* The monitor must have a clock that shows the actual time, including seconds.

**RQ-06**

* When the monitor detects that a component is offline (no heartbeat), it should write a message to the Incident Log (see below).

**RQ-07**

* There must also be a feature that lets the user view the Incident Log.

**RQ-08**

* The components should be colored as follows:

1. **GREEN** if it is online and all of its child components are online
2. **ORANGE** if it is online but one or more of its child components is offline
3. **RED** if it is offline

**RQ-09**

* In addition, the following textual information should be displayed.

1. Plant - fuel type, fuel level, maximum capacity, current speed, current power output, and percentage of total grip supply.
2. Substation - current load, and percentage of total grid load.
3. Feeder Line - current load of all consumers attached to the line, and percentage of total of all lines belonging to this line's substation.
4. Consumer - current load, and percentage of total load on this feeder line.
5. Summary Section - total grid production, total grid consumption (load), and variance.
6. Clock - current time, including seconds

## **Grid Controller**

**RQ-10**

* Plants - increase/decrease power production by adjusting the running speed of the generators.

**RQ-11**

* Substations, Feeder Lines, and Consumers - take them offline or bring them back online.

**RQ-12**

* The appropriate entity object should be updated. This is necessary so that the Monitor always has access to the latest data.

**RQ-13**

* The change should be written to the Incident Log. In the case of taking a component offline, an explanation should be included with the log message - is this in response to a problem elsewhere in the grid, or is it a simulation of a problem as part of a testing routing.

## **Consumer Simulator**

**RQ-14**

* Every D seconds (D is the data update frequency specified in the configuration), the simulator should create demand data for every consumer, and write this data to the appropriate entity objects.

## **Data Writer**

**RQ-15**

* Every H seconds, read the heartbeat of all grid components (from the entity objects) and then write this information to the database.

**RQ-16**

* Every D seconds:
  1. Read the production (output/supply) data from the Plant objects and write this to the database.
  2. Read the consumption (demand/load) data from the Consumer objects and write this to the database.

## **Incident Logging**

**RQ-17**

* Whenever a component goes offline or comes back online, this event should be logged to an Incident Database. The log message should include the component ID, the timestamp, and whether the incident was due to human intervention or to a (simulated) external event.

## **CRUD Features and/or Application Builder**

**RQ-18**

* A GUI will be provided that lets an authorized user add/edit/delete components.

**RQ-19**

* A grid and all of its components will be specified in a JSON file, and this file can then be loaded on startup. (So, to remove a component, for example, the user can change the JSON file and then reload it.)

## **Production and Testing Modes**

**RQ-20**

* When the application starts, it must be in either testing mode or in production mode. Each mode should have its own dedicated database and incident log, separate from the other modes.

## **Others**

**RQ-21**

* Live Usage Graph

**RQ-22**

* Web Workers (Timers in Background)

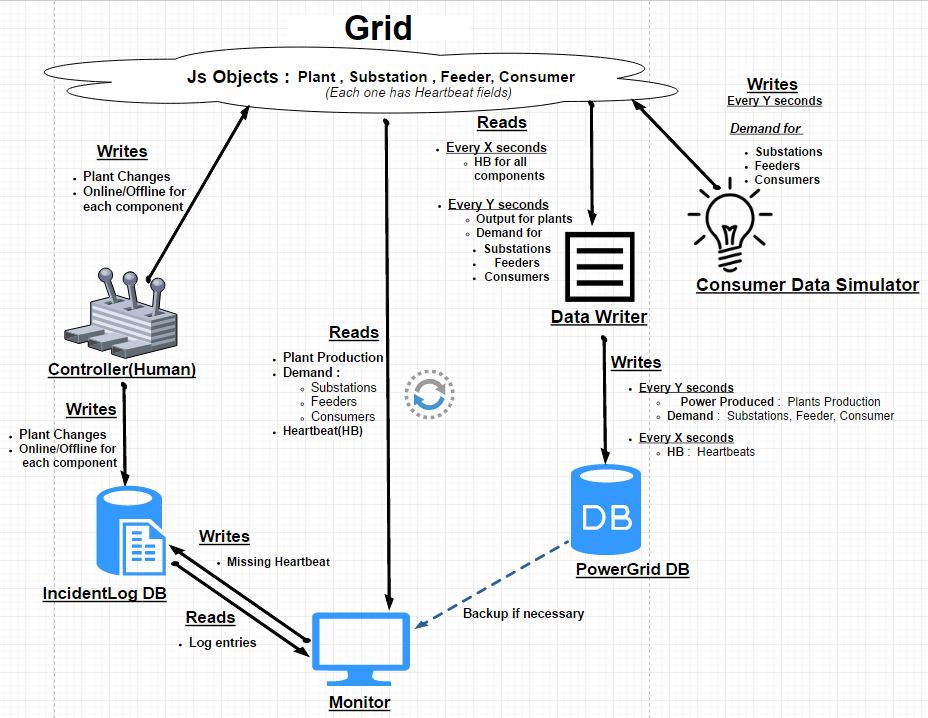
## **Putting It All Together**

**RQ-23**

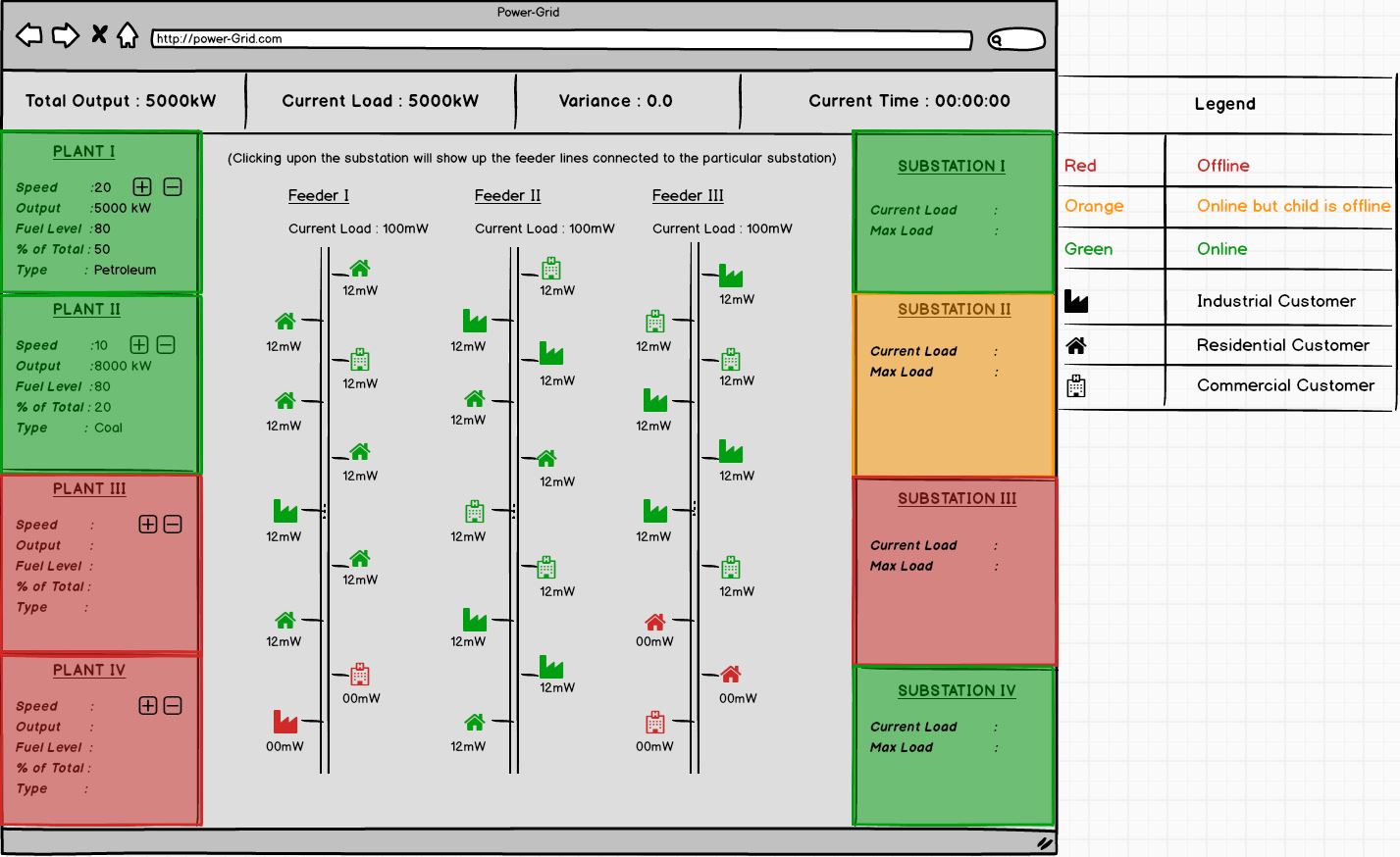
* A JSON file, “config.json”, that specifies:

1. Which mode (testing or production) the application should run in
2. The name of a server-side process to handle database access
3. The name of a server-side process to handle incident logging
4. The frequency of updates for power production and consumption
5. The frequency of status updates (heartbeats)

# **Grid Model Diagram**



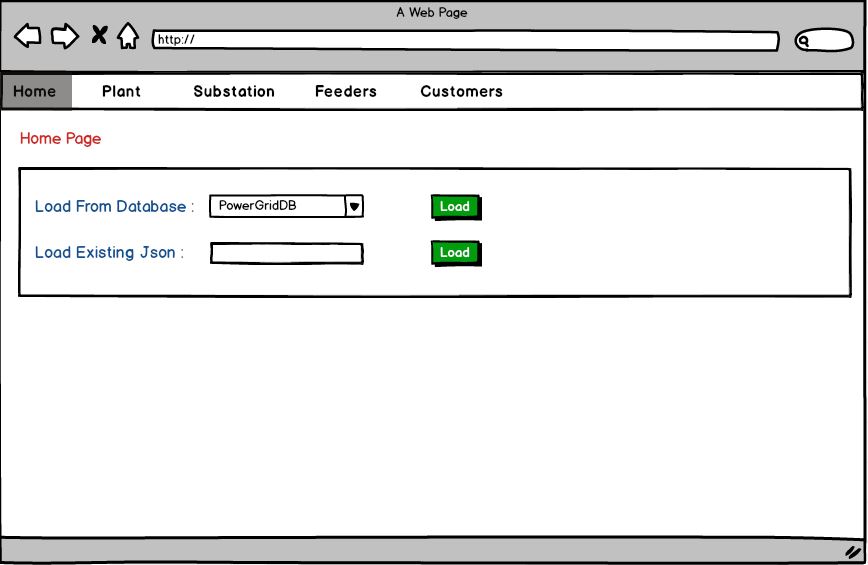
# **Power Grid GUI Mockup**



# **Power Grid CRUD Mockup**

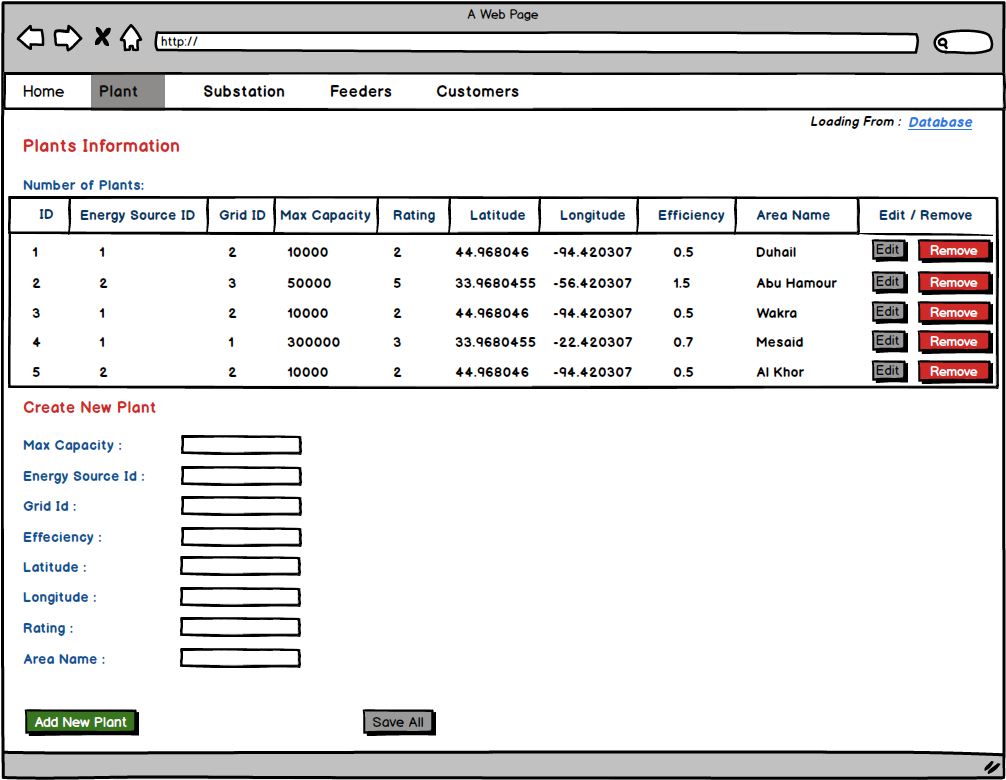
## **Home**

# 



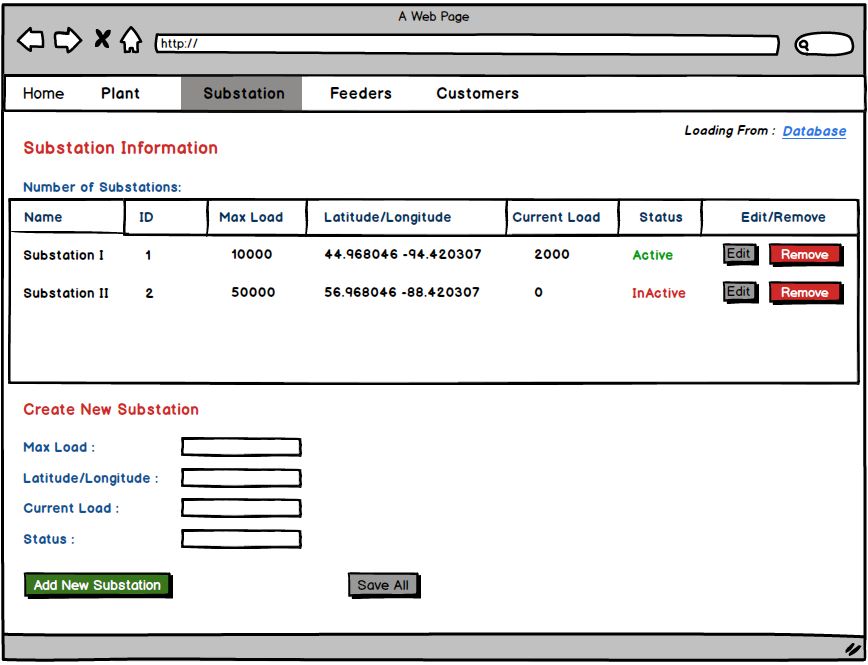
## **Plant**

## 



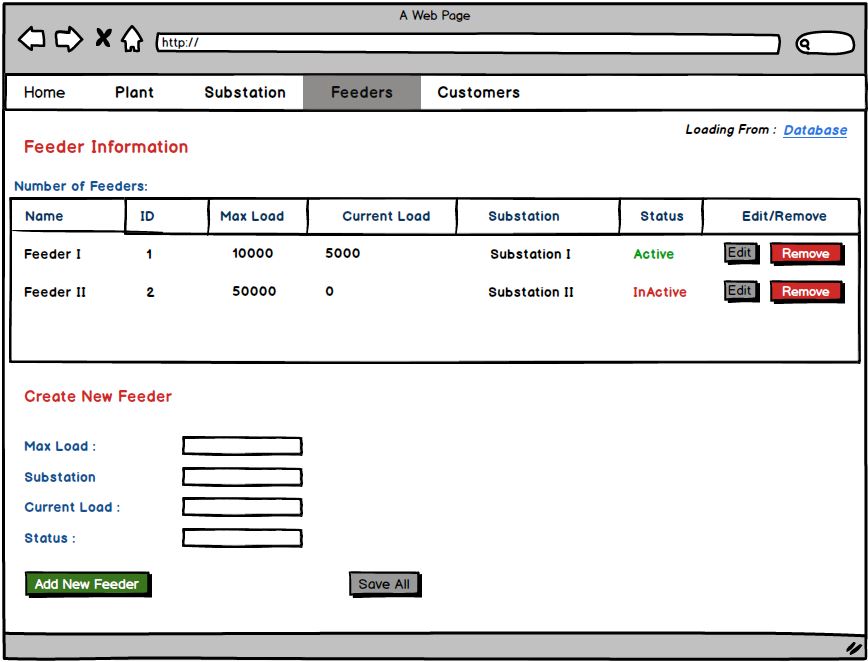
## **Substation**

## 



## **Feeder**

## 



## 

## **Customer**

