READ ME

Work Distribution:

This assignment work is submitted in the following manner:

- Database Design, ERP Module, MES Module → Khurshid Ali Qureshi
- Database Design, MES Module, SCADA Module → Umer Iftikhar

Software Used:

- Visual Studio (version 4.6.01038) as JavaScript editor.
- Notepad ++
- MySQL (version 5.7.11)
- Postman

npm Modules Used:

- request
- express
- mysql
- hypermedia-type
- node-uuid

Database Design

We have included the following tables in our schema:

Accounts:

userName	password	telephone	address
Ali	asd	465853853	8 C 50
Ateeb	fgh	465853859	8 C 51
Umer	jkl	465853860	8 C 52
NULL	NULL	NULL	NULL

Phones:

phoneName	keyboard	screen	frame	phoneCost
phoneA	kbd1	screen3	frame2	100
phoneB	kbd1	screen3	frame1	200
phoneC	kbd3	screen2	frame1	300
NULL	NULL	NULL	NULL	NULL

Customer Order:

id	userName	phoneName	quantity	totalCost	orderStatus
0325eea3-d965-4a99-8a53-ba5bbc3f9ba6	Ali	phoneB	10	2000	pending
078f1fd0-cc70-4bad-88cf-218fa95c97ae	Umer	phoneB	10	2000	pending
116a302b-1cf8-497b-9a6e-b0ca7b1a6fef	Ali	phoneB	17	3400	pending
22b103a9-2756-46cb-addc-b0217939cbf4	Ali	phoneA	18	1800	pending
3bb71c80-c8cd-41d0-849c-2585e73ac747	Ali	phoneA	18	1800	pending
45b84395-db4e-46c9-b4a3-1311f20f5312	Ali	phoneA	18	1800	pending
489cbb5c-9e02-4adb-b82e-dd71e86dde64	Ateeb	phoneA	19	1900	sent to production

Production Order:

id	customerOrder	productionStatus
0325eea3-d965-4a99-8a53-ba5bbc3f9ba7	0325eea3-d965-4a99-8a53-ba5bbc3f9ba6	processing
0325eea3-d965-4a99-8a53-ba5bbc3f9ba9	0325eea3-d965-4a99-8a53-ba5bbc3f9ba6	processing
195da56e-034c-4cf9-92d7-6170818d10ad	b06e7bd0-6c84-4d3c-9905-1a5b9ba5a30f	processing
db50e12d-ea0d-4d21-9b4e-3ede617a68e5	cf04db1f-381f-4210-aed1-4ba63d2459de	processing
NULL	NULL	

Purchase Order:

id	productionOrder	materialName	quantity	POStatus	supplier
b5e0f529-33ff-49c6-aa71-8ecf87e0deef	195da56e-034c-4cf9-92d7-6170818d10ad	kbd3	4	completed	kbd3 supplier
fa8e22b9-c249-41c1-8502-9806aaef1ade	195da56e-034c-4cf9-92d7-6170818d10ad	kbd2	6	completed	kbd2 supplier
NULL	NULL	NULL	NULL	NULL	NULL

Warehouse Stock

materialName	currentStock
frame1	10
frame2	10
frame3	10
kbd1	10
kbd2	16
kbd3	30
screen1	10
screen2	10

API Structure Description

ERP

Insert Customer Order

```
{
	"name": "add customer order",
	"method": "POST",
	"href": "http://localhost:8080/erp/customerOrder",
	"fields": [
		{ "name": "userName", "type": "varchar(20)" },
		{ "name": "phoneName", "type": "varchar(10)" },
		{ "name": "quantity", "type": "int" }
	]
},
```

Get Customer Order by ID or Username

```
{
	"name": "get customer order by id",
	"method": "GET",
	"href": "http://localhost:8080/erp/customerOrder",
	"fields": [
		{ "name": "userName", "type": "varchar(20)" },
		{ "name": "id", "type": "varchar(36)" }
	]

},
```

Insert Purchase Order

```
{
	"name": "add purchase order",
	"method": "POST",
	"href": "http://localhost:8080/erp/purchaseOrder",
	"fields": [
		{ "name": "materialName", "type": "text" },
		{ "name": "POId", "type": "varchar(36)" },
		{ "name": "quantity", "type": "int" }
	]
},
```

Get Purchase Order Status by ID

```
{
	"name": "get purchase order by id",
	"method": "GET",
	"href": "http://localhost:8080/erp/customerOrder/id",
},
```

Update Purchase Order Status by ID

```
{
	"name": " update purchase order by id ",
	"method": "PUT",
	"href": "http://localhost:8080/erp/customerOrder/id",
	"fields": [
		{ "name": "POStatus", "type": "text" }
	]
},
```

MES

Get Warehouse by Type

Fetch any already entered warehouse entry using its Type.

```
{
	"name": "get by type",
	"method": "GET",
	"href": "http://localhost:8080/warehouse/type"
},
```

Get Warehouse by ID

Fetch any already entered warehouse entry using its ID.

```
{
	"name": "get by id",
	"method": "GET",
	"href": "http://localhost:8080/warehouse/type/id"
},
```

Update Warehouse Using ID

```
{
	"name": "update by id",
	"method": "PUT",
	"href": "http://localhost:8080/warehouse/type/id",
	"fields": [
		{ "name": "category", "type": "string" },
		{ "name": "type", "type": "string" },
		{ "name": "quantity", "type": "int" },
		{ "name": "supplier", "type": "string" },
		{ "name": "productionOrder", "type": "string" }
	]
},
```

System Flow

The whole process is automated. We need to start all the files of ERP, MES and SCADA which are on following Ports.

ERP	9000
MES	8000
SCADA	2999

And run the Fastory Simulator on port 3000

Just add the customer order and the rest of the order is completes using in code API calls

The data flow is shown in the below flowchart.