Schemas:

User schema:

User\_ Schema {

Name : String,

Phone : Number,

Email : String,

Password : String,

Address : String,

District : String,

Pincode : Number,

State : String

}

Item Schema:

Item\_Schema {

Item\_name : { String, Enum : [

Shirt, T-shirt, Trousers, Jeans, Boxers, Joggers

] },

Item\_Quantity : { Number, Default :”0”},

Wash\_Type : { String, Enum : [

{ RegularWash , Price : 20}, { Iron , Price : 15 },

{ DryClean , Price : 25 }, { Pack , Price : 10 }

] },

Order : { type : Mongoose.Schema.Type.ObjectId , ref : “Order\_Schema” }

}

Order Schema:

Order\_Schema {

Order\_Id : { Number},

Order\_D&T : { Date() , Time()},

(date and time) timestamp

Total\_items : { Number },

Total\_Price : { Number },

Status : { String, Enum:[

Pending, PickedUp, InProgress, ReadyToDeliver, Delivered, Cancelled ], Default :”Pending” },

Store : { type : Mongoose.Schema.Type.ObjectId , ref : “Store\_Schema”},

User : { type : Mongoose.Schema.Type.ObjectId , ref : “User\_Schema” },

Address : { String },

Cancellation : Boolean

}

}

**Store Schema:**

Store\_Schema {

Store\_Id : { Number},

Store\_Name : { String },

City : { String },

Phone : { Number },

Address : { String }

}

1st User{

Order 1{

1. Item{

Wash types{ iron , regular wash }, quantity }

2. Item{

Wash types{ iron , regular wash }, quantity }

3. Item …

}

Order 2{

1. Item{

Wash types{ iron , regular wash }, quantity }

2. Item{

Wash types{ iron , regular wash }, quantity }

3. Item …

}