class Spares(models.Model):

    spare\_class = models.CharField(max\_length=100, default='')

    equipment\_class = models.CharField(max\_length=100, default='')

    pattern\_number = models.SlugField(max\_length=100, default='', blank=True)

    description = models.CharField(max\_length=500, default='')

    category = models.CharField(max\_length=20)

    critical = models.BooleanField(default=False, blank=True)

    # Location specific details to be stored in the database

    compartment = models.CharField(max\_length=200, default='UNKNOWN', blank=True)

    location = models.CharField(max\_length=200, default='UNKNOWN', blank=True)

    # Quantity specific details

    denomination = models.CharField(max\_length=20, default='')

    quantity\_authorised = models.PositiveSmallIntegerField(default=0, blank=True)

    quantity\_available = models.PositiveSmallIntegerField(default=0, blank=True)

    # Extra details regarding the spare

    authority = models.CharField(max\_length=100, default='D787J')

    page = models.SlugField(max\_length=100, default='', blank=True)

    line = models.PositiveSmallIntegerField(null=True, blank=True)

    remarks = models.CharField(max\_length=200, null = True,blank=True, default='')

class Issue(models.Model):

    # The issued spare

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    # User to which the item is issued

    username = models.CharField(max\_length=50, default='')

    quantity\_issued = models.PositiveSmallIntegerField(default=0)

    issue\_time = models.DateTimeField(auto\_now\_add=True, null=True)

    remarks = models.CharField(max\_length=200, default='')

class IssueList(models.Model): --TEMPERORY

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    quantity\_toreturn = models.PositiveSmallIntegerField(default=0)

    username = models.SlugField(max\_length=50, default='')

class Return(models.Model):

    # The issued spare

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    # User to which the item is issued

    username = models.CharField(max\_length=50, default='')

    quantity\_returned = models.PositiveSmallIntegerField(default=0)

    return\_time = models.DateTimeField(auto\_now\_add=True, null=True)

    remarks = models.CharField(max\_length=200, blank=True, default='')

class Survey(models.Model):  #temperory

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    quantity\_tosurvey = models.PositiveSmallIntegerField(default=0)

class PostSurvey(models.Model):

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    quantity\_surveyed = models.PositiveSmallIntegerField(default=0)

    survey\_number = models.SlugField(max\_length=50, default='NA')

    survey\_number\_date = models.DateTimeField(null=True)

    survey\_report\_date = models.DateTimeField(null=True)

    remarks = models.CharField(max\_length=200, blank=True, default='')

class Demand(models.Model): #TEMPERORY

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    quantity\_todemand = models.PositiveSmallIntegerField(default=0)

    survey\_entry = models.ForeignKey(PostSurvey, on\_delete=models.PROTECT, null = True)

class PostDemand(models.Model):

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    quantity\_demanded = models.PositiveSmallIntegerField(default=0)

    demand\_number = models.SlugField(max\_length=50, default='')

    demand\_date = models.DateTimeField(null=True)

    remarks = models.CharField(max\_length=200, blank=True, default='')

class Receive(models.Model): #TEMPEORY

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    quantity\_toreceive = models.PositiveSmallIntegerField(default=0)

    demand\_entry = models.ForeignKey(PostDemand, on\_delete=models.PROTECT, null = True)

class PostReceive(models.Model):

    spare = models.ForeignKey(Spares, on\_delete=models.PROTECT)

    quantity\_received = models.PositiveSmallIntegerField(default=0)

    receipt\_number = models.SlugField(max\_length=50, default='')

    receive\_date = models.DateTimeField(null=True)

    remarks = models.CharField(max\_length=200, blank=True, default='')

class SpareClass(models.Model):

    name = models.CharField(max\_length=100, unique=True)

class EquipmentClass(models.Model):

    name = models.CharField(max\_length=100)

    spare\_class = models.CharField(max\_length=100)

class Denomination(models.Model):

    name = models.CharField(max\_length=50, unique=True)

class Authority(models.Model):

    name = models.CharField(max\_length=100, unique=True)