**Smart Photo Service**

***(Misra Studio)***

Functional Requirements

Python Backend Requirement:

Model training, mainly working on Deep learning network (python + Django/Flask use)

**Agenda** collect (wedding) photos as a dataset that matches faces and classify their faces accordingly

Side face front face, faces in all direction must be recognized by the model

Java Backend requirement

Register as a host :

A unique code must be created that should be unique for every wedding request

That code must be delegated to all other members of family to register as a wedding member

Create a wedding host :

A container must be defined in which details for wedding must be given inside and one online wedding invitation should be there for personalize user experience

Creating wedding as a different entity:

link with Wedding Host

Wedding cannot exist without a Wedding Host

Register as a wedding member

-to register and then user must submit his or her photos for model training

Get his or her images

Once user face is taken, he can view his or her selected photos for them

Download link

One link is should be given download all his/her photos

**Requirement Specification**

**Wedding Host : (done)**

Wedding Host is created contains the following fields

Wedding Host Id

Host Email Id

List<WeddingObject> it holds (one Host can have multiple Wedding)

Wedding Host has one Wedding Object

Relation b/w Wedding Host Wedding Host -------- one to many ---------- Wedding

Permissions :

* Create Permissions
  + Wedding host can create a wedding Host Entity
  + Same Wedding Host can add new Wedding

**Wedding Object : (created by Wedding Host)**

Wedding Object unique ID (code)

Wedding Title

Wedding (main) Date

Wedding Events (list)

Wedding Host (foreign key)

List of Wedding Members (list one wedding can have multiple wedding members)

Permissions :

* Create permissions
  + Wedding cannot be created or manipulated at its own

**Wedding Member : \*\*\*\*\***

Unique Id

Member Name

Member Email ID

Member Relation

AttendsWeddingCode (list) -- initially null at creation (one member can have multiple Wedding)

Photos of him/her

Permsissions :

* Create permissions
  + Wedding Member can create its Entity
  + Wedding Member can register itself to a wedding
  + Wedding Member can add its photos to its Entity