Long Parameter List

Item class constructor has more than three or four parameters, but not all these parameters are really can be used. For example, a description or a maker may be omitted. Builder pattern can help to solve this problem.

```
public class Item extends Observable {
  public Item(String title, String maker, String description,
Bitmap image, String id) {
  }
}
```

Solution:

```
ItemBuilder itemBuilder = new ItemBuild();
itemBuilder.setTitle();
Item item = itemBuilder.build();
```

Or using method chaining (a fluent interface)

```
Item item = new ItemBuilder().setTitle().build();
```

Data class / Duplicate code

Item and Contact classes have only getters and setters, but they don't check parameters. Activities check parameters before creating objects. If the same class is used by many classes and validation rules are changed, all this code must also be updated.

Solution:

ItemList, ContactList, Item and Contact classes may validate data..