Capstone Assignment 2.3 part1

Code Smell 1: ***Feature Envy***

- The code smell can be found in ContactsActivity.java .

```
public boolean onItemLongClick(AdapterView<?> parent, View view, int pos, long id) {

    Contact contact = adapter.getItem(pos);

    // Do not allow an "active" borrower to be edited
    active_borrowers_list_controller.setContacts(item_list_controller.getActiveBorrowers());
    if (active_borrowers_list_controller != null) {
        if (active_borrowers_list_controller.hasContact(contact)) {
            CharSequence text = "Cannot edit or delete active borrower!";
            int duration = Toast.LENGTH_SHORT;
            Toast.makeText(context, text, duration).show();
            return true;
        }
    }

    contact_list_controller.loadContacts(context); // must load contacts again here
    int meta_pos = contact_list_controller.getIndex(contact);

    Intent intent = new Intent(context, EditContactActivity.class);
    intent.putExtra("position", meta_pos);
    startActivity(intent);

    return true;
}
```

- This method “onItemLongClick()” is more interested in the details of ContactListController class.
- The methods highlighted above belong to the ContactListController class.
- Thus implementation of “onItemLongClick()” method would be more appropriate in ContactListController class to avoid the dependency of ContactsActivity class on ContactListController class. This would resolve the code smell of Feature Envy.

Code Smell 2: *Data Class*

- The code smell can be found in Dimensions.java .

```
public class Dimensions {

    private String length;
    private String width;
    private String height;

    public Dimensions(String length, String width, String height) {
        this.length = length;
        this.width = width;
        this.height = height;
    }

    public String getLength() {
        return length;
    }

    public String getWidth() {
        return width;
    }

    public String getHeight() {
        return height;
    }

    public String getDimensions() {
        return length + " x " + width + " x " + height;
    }
}
```

- The “Dimensions” class contains only data and no real functionality.
- This class has only getter methods and nothing else.
- This indicates that it may not be a good abstraction or a necessary class.
- Some behaviors of other classes that manipulate this data could be better placed in this data class. For example, dimension validation in item class can be moved here.