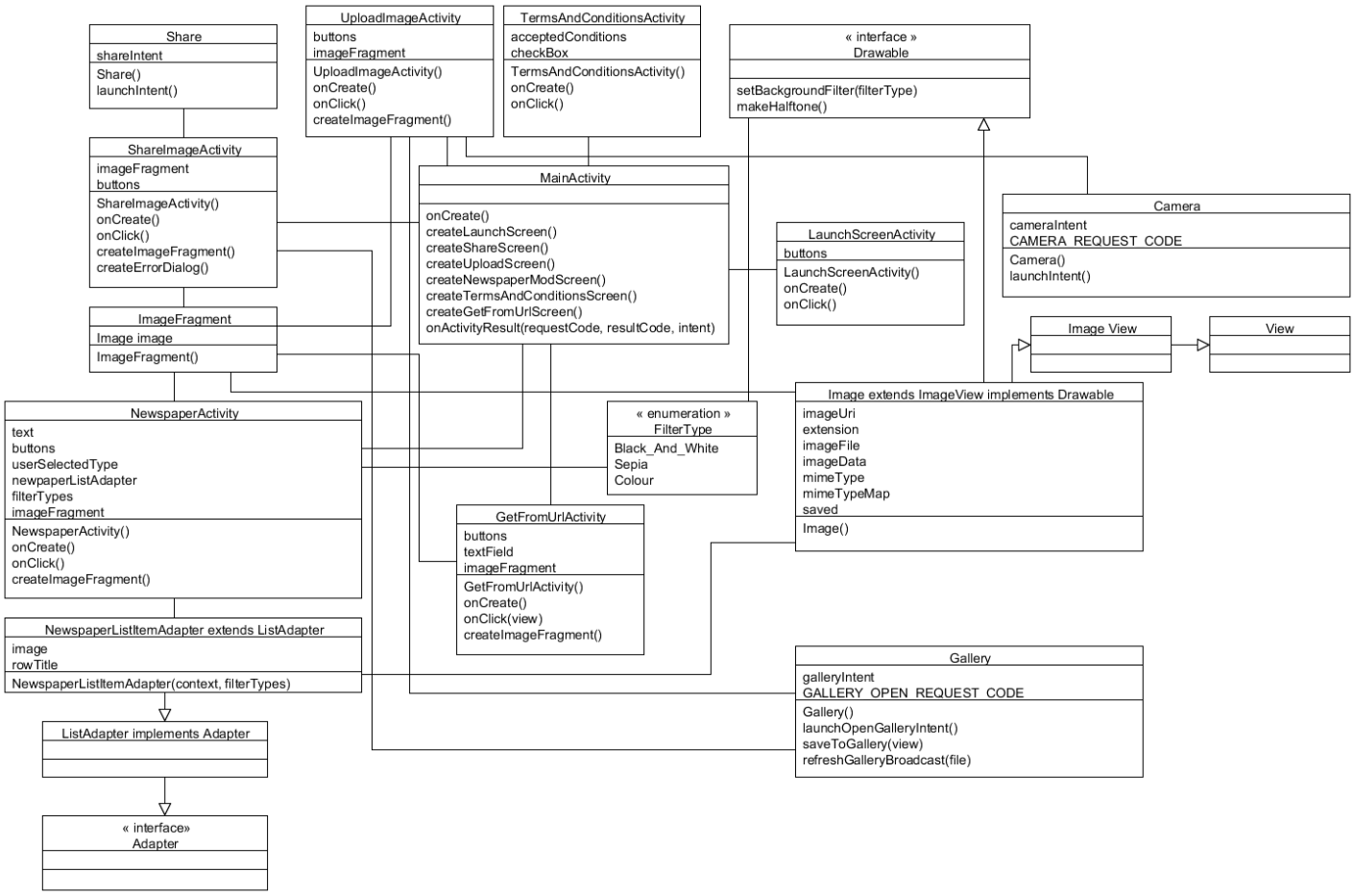
Conceptual UML Class Diagram

Provided below is the conceptual UML class diagram for the Ye Olde Times Android mobile application. The diagram shows all of the entities within the system in addition to the attributes belonging to those entities and the tasks that the entities are responsible for performing. Below the diagram is a comprehensive breakdown explaining all of the entities within the system and how they relate to one another. It is important to note when viewing the below conceptual UML class diagram that the accessors and mutators for each of the attributes in each class have been omitted for the sake of simplicity however they will be included in the final implementation of the application. As the names of the entities within the diagram have been concatenated into a single word following class naming convention, the names of the classes have been separated for easier readability in the explanation below (for example the "UploadImageActivity" class is described below as the "Upload Image Activity" class).



Main Activity

The Main Activity is the main entry-point of the application. It is responsible for calling the onCreate method which sets up all of the activities accessible from the Main Activity. It also contains 6 methods which create the various activities of the application, each of which represent a screen within the application. It also has an onActivityResult method which handles the result of any intents called from any activities within the application. Main Activity has a relationship with the Upload Image Activity, Terms and Conditions Activity, Get from URL Activity, Newspaper Activity, Share Image Activity and Launch Screen Activity as these are the activities that it creates in its 6 "create screen" methods.

Terms and Conditions Activity

The Terms and Conditions Activity contains two attributes: acceptedConditions and checkbox. AcceptedConditions is an attribute which records whether the user has accepted the terms and conditions statement. The checkbox attribute is an actual checkbox which is displayed to the screen for the user to click. The Terms and Conditions activity also contains a constructor, an onCreate and an onClick method. The onClick method is called on the click

of the checkbox while the onCreate method is called when the activity is created. The activity has one relationship to the Main Activity entity. This is because the Main Activity entity creates the Terms and Conditions Activity.

Upload Image Activity

The Upload Image Activity has two attributes, one called "buttons" which contains all of the buttons on the screen and another called imageFragment that contains an instance of the Image Fragment class. The onClick method inside of the entity handles all of the functionality to perform when these buttons are clicked. There is also a constructor, an onCreate and a createImageFragment. The createImageFragment method creates an Image Fragment such that the image to be uploaded can be displayed within the image view in the Image Fragment. Meanwhile the onCreate method is called on instantiation of the activity. The Upload Image Activity has relationships with the Image Fragment and Main Activity entities because it contains an instance of Image Fragment and is created by the Main Activity. It also has relationships with the Gallery and Camera classes because it calls the openGallery intent or the camera intent to obtain an so that it can be placed into the Image Fragment's image view.

Share Image Activity

The Share Image Activity contains a "buttons" attribute which contains all of the buttons on the screen and an imageFragment attribute which contains an instance of the Image Fragment class that is used to display an image to the screen. The onClick method inside of the entity handles all of the functionality to perform when these buttons are clicked. There is also a constructor, an onCreate and a createImageFragment method. The createImageFragment method creates an image fragment to the screen such that the image to be uploaded can be displayed. Meanwhile the onCreate method is called when the activity is created. The buttons on this screen allow the user to select a social networking service or email to share their image with. There is also a createErrorDialog method which allows for the creation of several error dialogs including one to deal with the situation where the user has not saved their image and another which alerts the user if they cannot share with a particular social networking service because they do not have it installed on their device. The Share Image Activity contains 4 relationships; one with the Image Fragment entity, one with the Share entity, one with the Gallery entity and one with the Main Activity entity. It has a relationship with the Image Fragment entity as it contains an instance of Image Fragment itself. The Share entity is used by the Share Image Activity to create a share intent for the image. The Share Image Activity also requires access to the saveToGallery method within the Gallery class and thus has a relationship with the Gallery class. Finally, it has a relationship with the Main Activity entity because the Main Activity entity creates the Share Image Activity.

Image Fragment

The Image Fragment contains an image attribute which is an instance of the Image class (which is essentially an ImageView with some additional attributes and methods). It contains a constructor in addition to 5 relationships to other classes in the diagram. It has a relationship with the Share Image Activity, Upload Image Activity, Get From Url Activity and Newspaper Activity because those activities have an instance of an Image Fragment within them. It also has a relationship to the Image class because the Image Fragment contains an instance of the Image class.

Newspaper Activity

The Newspaper Activity contains a text attribute which contains the text that the user wants to put into their 19th century style newspaper article and an imageFragment attribute which contains the image fragment used to display the newspaper image. It also contains the buttons used on the screen and an onClick method which handles what occurs when those buttons are clicked. It also has an attribute called userSelectedType which is the type of newspaper layout selected by the user. Additionally, it contains a newspaperListAdapter which is the adapter for the rows in the list view displayed in the activity. The final attribute that it contains is a series of filterTypes which are able to be selected by the user to colour their newspaper. The class also has a createImageFragment method which creates the Image Fragment for the screen. Additionally, it contains a constructor and an onCreate method which are called when the activity is created. The Newspaper Activity class contains 4 relationships. There is a relationship between the Newspaper Activity class and the Image Fragment and Newspaper List Item Adapter classes because the Newspaper Activity contains both an image fragment and makes use of the list item adapter to display the

different types of layouts selectable by the user. It also has a relationship to the Filter Type enumeration because it contains a series of Filter Types which denote the types of filters that the user should be able to select and a userSelectedType which is a particular filter type that the user has selected. Finally, the Newspaper Activity has a relationship with the Main Activity because it is created by the Main Activity class.

Get From URL Activity

The Get From URL Activity has a buttons, textField and imageFragment attribute. The buttons encompass all of the buttons in the activity meanwhile the textField is the field used by the user to input a URL to an image online. Finally, the imageFragment attribute contains the image fragment which is contained within this activity. The entity also contains a constructor, an onCreate and an onClick method. The onClick method handles all of the button clicks meanwhile another method called "createImageFragment" creates the image fragment for inclusion within this view.

Launch Screen Activity

The Launch Screen Activity has some buttons, a constructor, an onCreate method and an onClick method which determines what to do when the buttons on the screen are clicked. The Launch Screen Activity is displayed upon the user first opening the app and serves as the "home screen" of the application. It has a single relationship to the Main Activity class as the Launch Screen Activity is created by the Main Activity class.

Newspaper List Item Adapter

The Newspaper List Item Adapter is an adapter that extends from the base adapter for list items known as the List Adapter. It has a single attribute that is an Image (image view) which displays the appropriate layout type for the newspaper. It also contains a rowTitle for the list item row that labels the image in the row.

List Adapter

The implementation of the List Adapter class has not been provided as it is a part of the Android API. In summary, the List Adapter class is the low level adapter for the items in lists created using the Android API. The List Adapter also implements the Adapter interface which is again directly taken from the Android API.

Adapter

The Adapter interface acts as a bridge between an AdapterView (in this case a List Adapter) and the underlying data for that view. Again, the implementation of this class has not been included in the diagram as it is a class that is part of the Android API.

Image View

The Image View class is a class that is a part of the Android API. In summary, it displays an image and handles functionality to do with images such as scaling and tinting. Again, the implementation of this class has not been included in the diagram as it is a class that is part of the Android API.

View

The View class is a class that is a part of the Android API as well. In summary, it acts as a very rudimentary building block for user interface components. It is primarily responsible for drawing components to the screen. Implementation has not been provided for this class as it is a part of the Android API.

Camera

The Camera class contains a cameraIntent and a CAMERA_REUQEST_CODE attribute. The cameraIntent contains the intent to launch the camera and obtain image data once the camera intent has finished. Meanwhile the CAMERA_REUQEST_CODE is used to process what is returned by the camera intent in the onActivityResult method that receives the result of the camera intent call. It also contains a constructor and a launchIntent method

which launches the camera intent. It contains a single relationship to the UploadImageActivity class as the Upload Image Activity allows the user to obtain their image from their camera in addition to their gallery.

Gallery

The Gallery class contains the galleryIntent and GALLERY_OPEN_REQUEST_CODE. The galleryIntent contains the intent used to open the gallery while the GALLERY_OPEN_REQUEST_CODE contains the request code used to open the gallery intent. The Gallery class also has a constructor and methods to launch the gallery intent to open the gallery (launchOpenGalleryIntent), save an image to the gallery (saveToGallery), and send a broadcast to refresh the gallery (refreshGalleryBroadcast). It has a relationship to the Upload Image Activity class because the activity requires access to the gallery to load an image. It also has a relationship to the Share Activity class as Share Activity requires access to the saveToGallery method to save the image back into the gallery after it has been manipulated.

Image

The image class contains the imageUri (a Uri which points to the image's location), an extension for the type of image that is being displayed, an imageFile containing the original image loaded in, the imageData containing the data of the image in bytes, the mimeType (a further type which is used in the share intent to determine the type of image being shared), a mimeTypeMap used by the share intent and finally, a saved flag which determines whether the image has been saved or not. As the Image class is essentially an extension on the Android API's Image View class, it is used to display images in the application within the Newspaper List Item Adapter and the Item Fragment. Thus, the Image class has relationships to the Newspaper List Item Adapter and Item Fragment classes.

Share

The share entity contains a shareIntent attribute which is the share intent that can be called upon to share an image from another activity. It also contains a launchIntent method which launches the intent. It has a single relationship with the Share Image Activity class as this activity is the sole activity that requires the share intent.

Drawable

The Drawable interface contains a setBackgroundFilter method which sets the background filter of an image to a specific filter. It also contains a makeHalftone method which would make a given image into a halftone image. As many different types of halftone images could be included in future iterations of the application, the Drawable interface was determined to be necessary to ensure that these additional types of half-toning techniques could be easily included by implementing the Drawable interface in other image view classes. It has a relationship to the Filter Type Enumeration because it makes use of the enumeration in determining which filter was passed into its setBackgroundFilter method.

Filter Type Enumeration

The Filter Type Enumeration is an enumeration which includes the different types of filters available for the user to select from. It is used by both the Newspaper Activity class and the Drawable interface and thus, this is why relationships are shown between the Filter Type Enumeration and these two entities.

Domain Dictionary

Described below is the Domain Dictionary that has been included to explain all of the terms specific to the design of the Ye Olde Times application. Additional terms have been included that may relate to more technical concepts referred to in the design.

Activity: A single operation that the user can perform within a screen. They usually interact with the user and comprise of a user interface. The user interface may comprise fragments to display certain components of the user interface or may contain raw widgets.

Adapter: In the context of Android mobile application development, an Adapter provides access to data items within a view.

API: Application Programming Interface. An API specifies how different components of a software application interact with one another.

Broadcast: A broadcast is a call sent out to all applications interested in listening to the broadcast. A broadcast will sometimes ask a specific application to perform a specific action upon receiving a broadcast, however it can also broadcast to several applications in the situation that several applications must perform functionality on receiving the broadcast.

Class: An entity within the system.

Entity: A term used interchangeably with the term "class" to mean a component of the system, represented as a rectangle with a name, list of attributes and list of methods as displayed in the above conceptual UML diagram.

Enumeration: A collection of items which fully describes the collection in which it resides. For example, an enumeration could be the primary colours, which would comprise of red, green and blue.

Fragment: A piece of the application's user interface that can be placed within an Activity. It is a component that can comprise an entire screen or just a section of a screen that is used over and over again in different activities or fragments.

Image View: A view in which images can be placed. An image view allows a user to see an image on their device's screen.

Interface: An entity which comprises several methods which must be implemented by any class which inherits from the given interface. The purpose of having an interface is to ensure that many different classes can have different implementations of particular methods. It also ensures that every class that implements the interface must implement the given methods in some manner.

Intent: A description of an operation to be performed. An intent can open an activity or it can launch other applications on the user's device (for example: the camera application or the gallery application).

Method: An activity or responsibility of a particular class. Methods provide functionality that classes are responsible for performing.

Request Code: A code sent by an intent in order to be tracked once the intent finishes. It allows for data returned from the intent to be obtained upon checking for the request code.

Sharing: Sharing refers to providing data to another application. In the context of the Ye Olde Times Android mobile application, sharing refers to sharing images with social networking applications in addition to email on the user's device.

UML Diagram: Unified Modelling Language Diagram. A UML diagram describes all of the entities in the system and how they are related. The UML diagram provided in this document is a conceptual UML diagram which means

that specific details pertaining to the implementation of the application have been omitted (such as the type of attributes in each entity, the types of parameters passed in to method calls and the return types of method calls).

Uri: Uniform Resource Identifier. A Uri is an address to points to some data.

Widget: A component of the user interface that allows the user to view content or interact with content. Some examples of widgets include buttons and image views.