Team Name: Insert Team Name Here
Members: Daniel Anderson (d563a309@ku.com), Brandon Caudell (b851c341@ku.edu), Zach Williams (z035w745@ku.edu), Dustin Wendt (dustinwendt@ku.edu), Gabby Gasser (grgasser@ku.edu), TJ Barclay (tjbarclay@ku.edu)
Team Meeting time: 4:00 Wednesday
Lab Meeting time: 2:45 Wednesday
Contact: Brandon Caudell (b851c341@ku.edu)
Project Sponsor: None

Project Description:
Our project is to create an Android and/or VR application that allows for virtual graffiti. Users will be able to walk around in a simple augmented reality view, leaving publically-viewable artwork. For instance, a user might go to Eaton, and draw “Engineering Rocks!” on the wall. A second user should be able to then go to Eaton and look at the same wall (through either a VR product or the camera’s phone) and see the drawing. The social aspect of this application could be very appealing, as it allows a collection of users to turn a public space into an ever-changing display. From this basic idea, we could branch out in a number of different functionality directions depending on what we think might be valuable to a user base.

Project Milestones
1. Our first milestone will entail a user being able to walk to a location, drop a text message, and other users will receive a notification that there is a message nearby when they are within some radius of it. The notification should also read out the contents of the text message. The distinct functionality bits for this milestone are:
   a. Geo-location of the user
   b. User interface for entering a text message
   c. Having a storage location for the location-message pairs
   d. App interaction with the aforementioned storage location
   e. A periodic check in the app for the user being in range of a message
f. Interface to show the text from the message, and possibly its location as well
g. Create documentation for the application and provide general framework for future documentation

2. The next milestone will expand on the first by changing the message from textual to an image. This will require the following changes:
   a. Data storage must now be able to store image data, and the interactions with it will likely need to be changed due to higher data transfer volume
   b. Allow the app to open the user’s camera and upload an image
   c. Display the nearby image (still in a static form -- no AR yet)
   d. Add documentation

3. Our third milestone consists of taking the images from the previous one and processing them to eventually render in an augmented reality environment, the following pieces of functionality being key:
   a. Reading of the user’s gyroscope and compass data
   b. Calculation of the 3D orientation of the phone from the above data
   c. Calculation of distance between user and image (for image scaling)
   d. Determination of how much (if any) of the image is in the user’s field-of-view
   e. Clipping and transformation of the image to meet the expected perspective of the user
   f. Overlaying the transformed image on top of the live camera feed, creating AR
   g. Add documentation

4. Our fourth milestone is to use the transformed image from the third milestone to overlay the image onto the user’s camera feed to simulate augmented reality.
   a. Implementation of a “user complaint” system to help filter out inappropriate content from the app
   b. Add documentation

5. The fifth milestone would be fairly small by comparison. Once the previous one is implemented, we would want to create a drawing interface so that the uploaded images
are no longer photographs from the camera, but artwork drawn on the touch screen. This will also be reflected in documentation

6. Once the core functionality is implemented, there are a number of different expansions to the app that we can work on, time permitting:
   a. User approval or disapproval of artwork
   b. Ability to “follow” artists who the user likes, and see where more of their artwork is located
   c. A “private group” system, allowing artwork to optionally be private to within the group it was posted in
   d. Incorporate a virtual reality app that pairs with ours, perhaps acting as a kind of dynamic showcase of artwork around the world
   e. Additional drawing methods:
      i. Holding the phone as a paintbrush, tracking movement using the accelerometer, and translating that into an image
      ii. Using the phone as a “spraycan” to paint an image
      iii. Etc
   f. Add additional documentation

• Estimated completion date for each milestone
  o 1. October 24, 2016
  o 2. November 17, 2016
  o 3. December 16, 2016
  o 4. February 14, 2017
  o 5. March 17, 2017
  o 6. April 20, 2017
Project Budget

The cost of the project is still being determined. There are two potential sources of cost: server hardware/operation, and VR tech should we decide to go that route. We will certainly need some server architecture to host the data storage and manage interfacing with users. We are currently using a trial of Amazon Web Services, so currently no cost is involved. If we write too many times to the database within a month, then we will start to be charged. This charge is variant on how much we use the database. If we opt for a VR setup, then we may or may not need to purchase hardware for it. We are aware that KU does have a reasonable amount of virtual reality equipment maintained by Professor Gill, and so we may be able to use some of that. This phase of development would not be until much later in the lifecycle of the project, so we have time to look into this before diving in.

Work Plan

We have essentially three (fluid) roles in this project. We will have two front end developers (Zach and Gabby), two VR/image specialist (TJ and Brandon) and two database/back end developers (Dustin and Dan)
These roles are somewhat fluid, and all of us will be participating in documentation and research.

Github link

- https://github.com/bcaudell95/VirtualGraffiti
Preliminary Project Design

Graffito, our virtual graffiti application, has to have some core, basic functionality. As a minimal viable product, Graffito needs to be able to take in media from a user (a “sketch” or drawing), save that media to a geographic location, and other users will have the ability to visualize the “sketches” that have been saved to that location as well as add their own. Due to the ease of portability offered by Android development as well as the ability to add to the Google Play store, we have decided to develop in an Android environment.

Breaking our minimal viable product down into components, the software and application design comes to light. Starting with the ability to add media to a specific location involves various software requirements. Once access to the camera and GPS location is enabled by the user, our applications comes in and allows the user to add media. Right now, for our minimal viable product, the media a user can add depends on the milestone we are at. At the initial milestone, we are going to allow the user to add text. Next, we will have the user add an image. Our final iteration will be to allow the user to leave an original drawing in a location. This will involve an evolution of our database to handle not only text and geographic location, but also original imagery as well. Along with this, we need to be sure we calculate an accurate location for the user. Right now, we take an average of three locations to gain an accurate measure of where our user is to best ensure their “sketch” makes it to the proper location. The location and sketch are then saved to the database.
Another interaction we need to consider as we design our application is when a user nears a location with a sketch. As a user nears locations with sketches, the user can receive notifications there are sketches nearby. The GPS will update the location, and if there are, in fact, sketches, a message will be sent to the user to open the application and see the art in the area. One of the main concerns with this notification strategy is the strain on battery life. As with applications that consistently update GPS, like Pokemon Go, the user base gets frustrated if an application seemingly uses too much battery for no apparent reason. A fix we have considered is reducing the number of times GPS location is updated, so, rather than every minute or so, check location every hour or so.

The chart below illustrates a general overview of how the core components of our Android application interact with each other. This overview does not consider the extraneous goals we have for our project, but focus on the interactions described above. As we move further through
our milestones, this chart will become increasingly more complex with various components being added.

Beyond the core functionality of our application, we have other goals we wish to achieve with design considerations of their own. We want our application to be social, since graffiti itself is such a public form of art. This involves a user onboarding process. We will have users sign up through their Google accounts, since this will ease some of the workload on us. We want to focus on the experience of the application rather than creating a user creation flow from scratch which would involve passwords and authentications.

Another design consideration with relation to the social aspects of our application involve figuring out how to allow users to interact with the graffiti and sketches they see around their area. This also involves a discussion on the longevity of sketches in an environment. Graffiti in the real world gets painted over when it is not popular enough or is in an undesirable location, or is not great to look at.

We are still considering the design implications of the various ways we can take these ideas. Initially, when a user submits a sketch, they will be able to set a minimum time for the sketch to be public. It will be displayed for that set amount of time for the community to vote on. To have
art stay beyond the minimum threshold, users would upvote the content. More popular content (sketches) will have a longer lifespan in the location they were initially drawn in. Again, this is how art works in the real world. Popular art has better longevity than a page colored in a coloring book.

Along with this, we are also considering what will happen if a user does not have access to GPS location services. This could be due to slow loading times or being in a location where GPS signal does not reach (such as caves or subterranean basements). We want to ensure in our design that the issues regarding GPS come from a pure lack of signal, rather than a flaw in our own design. Along with this, we cannot create GPS signal where there isn’t any to begin with. Rather, we can take the approach of Pokemon Go and educate our users when they initially open the application on where to best place their sketches. This would be letting the user know to not place graffiti in basements or caves or other areas where GPS does not reach.

When going beyond our core functionality, the design and software considerations become numerous and as we continue on our development path, we will encounter more design decisions that will need to take place. We will have to consider whether or not to open the camera view when we initially open the application, and what the options are when the app initially opens. Along with this, if we end up completing our Android application, we will also have to consider how to refine our design, if we wish to do so, to be a VR application along with a mobile application. With the creation of an app for VR a whole other set of software design considerations are raised.
Ethical and Intellectual Property Issues

1. Misuse of app

   - One of the main ethical and IP issues we will face in the creation of our application will be the reproduction of artwork by individuals without proper authorization or references. We will have to seriously consider if we are willing to police the artwork produced by our application. One way to help ease this issue is to have a flag option on each piece of art that is published on our app. Users who see the art and see that it is stolen can flag it as such and from there, we can take down the art. The main goal is to embrace an open community of virtual graffiti artists while maintaining the integrity of individual artists and their work.

   - Inappropriate content is a possible ethical issue that we would face. Our app being used for anything offensive or illegal is strictly not allowed by our community guidelines, which will be expanded upon in the github readme. This issue is addressed in our 4th milestone with the implementation of a user complaint system. Users will be able to flag sketches that they view as inappropriate and they will be marked to be reviewed and/or removed.
• There is also a possibility that our app could be abused as an advertising service. These actions will also be not allowed in our terms of service and accounts that have been repeatedly reported for advertisements will be barred from service.

2. Naming conventions we use in our app are ™

• When it comes to intellectual property when creating a brand new application, we want to consider the phrasing we use and the names we use for our application to ensure we are not breaking any sort of trademarks or copyrights. We do not want to copy ideas or names since none of us on a college budget, can afford the legal struggle. Beyond this, groups of marketers and advertisers worked hard to employ the verbiage they utilize to describe their functionality and we do not want to take that from them. At this current juncture, it will not be a problem, as we have attempted to research all the names and ideas we are implementing to be sure they do not break any copyright or trademark and we will continue to do this research and alter course if necessary.

Change Log

1. Our github link has been changed, because we were initially having issues with importing gradle files and getting the project to work in Android Studio.
2. We moved target date of first milestone from 10/17 to 10/24 to line up with our meeting.
3. We added documentation to milestones now that we have further refined and fleshed out our expectations of each.
4. We pulled back our MVP definition one level to more accurately reflect the distinction between minimal viable product and our target product.