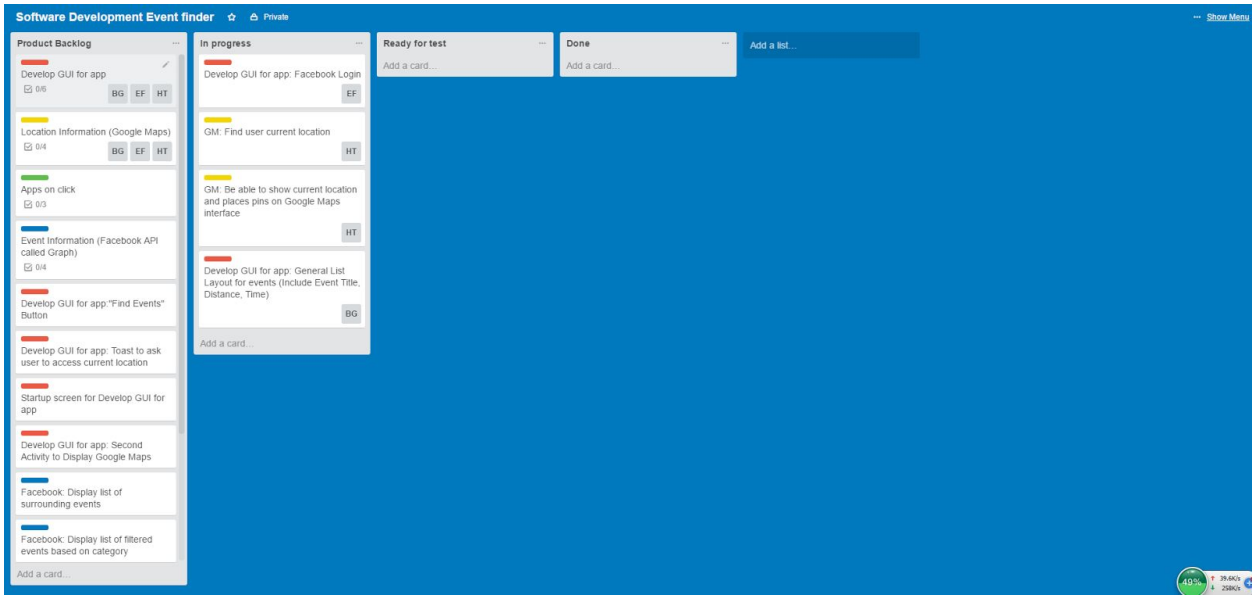
Milestone 2 Project Tools & Agile Methodology Summary

Project Management Tool

-Trello



Project Requirements

- Functional Requirements
  - Find the user's current location - 100 (Karros)
  - Connects user's facebook to app - 101 (Eli)
  - Populate list of surrounding events from facebook - 102 (Ha)
  - Create GUI that has button to find events - 200 (Bryan)
  - Create a google map GUI- 201(Jay)
  - Display pins of each event and your location on Google Maps - 202 (Eli)
  - Send notifications of upcoming events - 203 (Eli)
  - Display event distances within adjustable radius up to at least 25 miles - 105 (Eli)
  - Able to add filters - 107 (categories, distance) (Jay)
  - Disconnect from app - 108
- Non-functional Requirements
  - Populate surrounding events in under 30 seconds - 103 (Bryan)
  - Up to date events & locations - 104 (Ha)
  - App should be running in the background - 106 (Karros)
  - Keep user logged into Facebook and connected to app even after closing the app - 109 (Karros)
  - Database for facebook events details (longitude, latitude, titles, description, user preference) - 110

Project Plan

We are planning on doing four sprints, with the first sprint being the three weeks leading up to spring break. After spring break, we will be doing three more sprints each at a duration of two weeks. Tasks in each sprint will be decided at the end of the previous sprints and modified according to success of previous sprints. For this first sprint, the team members discussed their chosen tasks during the meeting along with the due date at which the tasks will be completed. The meeting minutes can be seen below.

Agile Methodology Conclusion and Summary

Elijah Fisher: I plan on implementing a basic startup screen and an interface for entering username and password information for Facebook connectivity. In order for this information to be processed Facebook integration will need to exist but I can make the window for the input without the input necessarily being used for the time being. This will be done by spring break (March 24).

Zhi & Karros Huang: Our main responsibility is to create a database and pull Facebook events information onto that database. By our third week, we will have learned the basics of how to use Facebook's API and MySQL. With this knowledge we will access basic Facebook Event information and store it into the database. Due March 24.

Bryan Gonzales: For this first sprint, I will be working on the general GUI for displaying the events and the format in which the events will be displayed once pulled from Facebook. The goal will be to get the basic outline of what the app will look like without the function of clickability. Testing of this will occur once we start integrating the data from Facebook and Google's API. This feature due date will be at the conclusion of the sprint, March 24th.

Ha Tran: I will be working on setting up basic location data from the Google Maps API, including current location and details about nearby locations such as name and location. I will also be working on displaying these locations as placed pins in the Google Maps interface. Due March 24.

Sprint Retrospective:
We don't necessarily have a retrospective for this meeting since we are going into our first sprint. However, our team discussed potential obstacles and challenges that could occur in the coming sprint and what steps can be taken to combat these challenges. We also discussed what has gone well so far in the planning stages of our project and what we can do as a team to increase efficiency of communication and collaboration.

User Cases

***Note:** 100 series IDs are back end functions; 200 series IDs are front end functions

ID:	**100**
Title:	Acquire User's geolocation data.
Description:	We will be using Google Map's API and geolocation
Primary Actor:	App
Preconditions:	App just opened, app prompts user to use their location
Postconditions:	The app has your Geolocation coordinates and stores it in the app.
Main Success Scenario:	App contains your geolocation data, that's to be used by other functions.
Extensions:	This data will be used in every function that requires a location.
Frequency of Use:	This will be used every single time the user opens the app.
Status:	Backlog
Owner:	TBD
Priority:	High

ID:	**101**
Title:	Connect User's Facebook to app.
Description:	This will connect the user to Facebook allowing us to use the api with information of their account.
Primary Actor:	App
Preconditions:	App has been just opened.
Postconditions:	User is logged in and the app is ready to be used.
Main Success Scenario:	Username and Password prompt pops up. User logs in. Prompt closes and user is notified that they are logged in. Main UI is loaded.
Extensions:	Will tell the user if they fail to log in and let them try again.
Frequency of Use:	Every time the app opens.
Status:	Backlog
Owner:	TBD
Priority:	High

ID:	**102**
Title:	List of Surrounding Events
Description:	The user selects a “Find Events” button that will then display a list of surrounding events based on the user’s location.
Primary Actor:	User
Preconditions:	App just detected and displayed the user’s current location.
Postconditions:	App displays a list of surrounding events based on current location with event info, which may include event title, date, time, and/or distance. User should be allowed to select event for more details.
Main Success Scenario:	After clicking “Find Events” button, app will display list of surrounding events.
Extensions:	If user does not have adequate data or wifi to find geolocations, app should display a notification letting user know that connection unavailable.
Frequency of Use:	Whenever the user wants to find surrounding events
Status:	Backlog
Owner:	TBD
Priority:	High

ID:	**103**
Title:	Start to finish app execution in under 30 seconds.
Description:	The basis for this use case is that we want the user to be able to open the app and find the events in under 30 seconds. Keeping app execution time under 30 seconds ensures that our app will be simpler than searching for events through the Facebook app.
Primary Actor:	Software
Preconditions:	The precondition is before the user has opened the open.
Postconditions:	The postcondition is when the app has retrieved and displayed the events.
Main Success Scenario:	Once the user opens the app, they will be prompted to log into Facebook (if this is the first time using the app), the app will take the location based on the radius setting, and then be taken to the interface with the map and button to find trigger retrieval of the events. Once the user has pressed the button, the events and locations will be displayed on the map as pins.
Extensions:	The exceptions in this case would be if there are problems with connectivity to Facebook or Google's API. This would case app execution to run longer that the proposed 30 seconds.
Frequency of Use:	Every time the user opens the app.
Status:	Backlog
Owner:	TBD
Priority:	High

ID:	**105**
Title:	Display event distances within adjustable radius
Description:	The app will display events within a radius up to at least 25 miles which can be adjusted.
Primary Actor:	The user will change the radius and the Facebook API will find the events within the radius.
Preconditions:	The app will be displaying locations with a previously set radius.
Postconditions:	The app will display locations within the specified radius.
Main Success Scenario:	The user will change the radius. The app will then use the Facebook API to find the events in the new radius and then display them.
Extensions:	If there are no locations within the radius, the user is notified of this.
Frequency of Use:	This will be used whenever the user changes the value of the radius or every 30 seconds.
Status:	Backlog
Owner:	TBD
Priority:	Medium

ID:	**106**
Title:	App runs in background
Description:	Keep app running in background to keep relevant data (geolocation, facebook events) constantly updated and send notifications.
Primary Actor:	User
Preconditions:	App is opened as main window on phone
Postconditions:	App is hidden in the background but still running
Main Success Scenario:	When user hides the app, the app will continue only running minimal processes such as updating relevant data, and sending background notification. When the user reopens the app, location is already up to date and only google map pin location needs to be updated.
Extensions:	[Describe all the other scenarios for this use case - including exceptions and error cases.]
Frequency of Use:	Whenever the user hides the app
Status:	Backlog
Owner:	TBD
Priority:	Medium

ID:	**107**
Title:	Ability to add filters
Description:	This will help you customize your preferences in order to narrow down your search results
Primary Actor:	User
Preconditions:	Apps fully functional, have user's current location
Postconditions:	Displays customized search results
Main Success Scenario:	Utilize user's location to get the best customized result for user.
Extensions:	No events match the filter
Frequency of Use:	Whenever user want to do a customized search.
Status:	Backlog
Owner:	TBD
Priority:	Low

ID:	**109**
Title:	Keep the user logged into Facebook and connected even after closing the app.
Description:	Users will not want to keep resigning back into facebook every time the app is closed, therefore we must store this data even after app is closed.
Primary Actor:	User
Preconditions:	User connects facebook.
Postconditions:	App is closed and facebook memory is saved.
Main Success Scenario:	User will log into facebook and the memory is stored in cache, even upon closing the app until user signs out of the app.
Extensions:	When user is disconnected from app, do not perform this function.
Frequency of Use:	As long as the user's facebook account is connected they will stay logged in.
Status:	Backlog
Owner:	TBD
Priority:	Medium

ID:	**110**
Title:	Database
Description:	Database will contain all the information pulled from facebook events, then the data will be pulled from the database and put onto Google maps.
Primary Actor:	User
Preconditions:	A stable internet connection and computer to host the server.
Postconditions:	A database containing all necessary information required to make our app functional.
Main Success Scenario:	A computer will be setup to be hosting a database. Then when the app is performing its functionality, it will prompt Facebook to put all of its events in the database, and then the Google Maps will grab that information, such as event title, time, and location, from the database and display them on its map.
Extensions:	Database is empty b/c no events are found. Create toast saying no events around.
Frequency of Use:	Every single time the app is running.
Status:	Backlog
Owner:	TBD
Priority:	High

ID:	**200**
Title:	Create GUI that has button to find events.
Description:	This feature is the implementation of the button which puts into action the retrieval of the events.
Primary Actor:	User
Preconditions:	The app at this point will running, and the location of the user will have already been taken and displayed on the Google maps interface.
Postconditions:	After the button has been pressed, the system will then jump to the retrieval of the events.
Main Success Scenario:	The flow of events is simply telling our system to retrieve the events by the press of a button.
Extensions:	The main error for this functional piece would be the scenario where the user hasn't connected his/her Facebook to the app. Other errors would include: connectivity problems, or troubles on the Facebook or Google side.
Frequency of Use:	This will be used everytime the user is ready to retrieve nearby events.
Status:	Backlog
Owner:	TBD
Priority:	Low

ID:	**201**
Title:	Create a google map GUI
Description:	This is a GUI that user open our apps, it will display in a map of the user current location
Primary Actor:	App
Preconditions:	Android phone has GPS enabled, user log into our apps or give permission for us to send notification in the background
Postconditions:	User will be able to view his current location in google map
Main Success Scenario:	User open up our apps,display google map format of user's current location
Extensions:	User doesn't have wifi connection or data usage at the moment, app will display an error message.
Frequency of Use:	Every time user opens app
Status:	Backlog
Owner:	TBD
Priority:	High

ID:	**202**
Title:	Display pins of each event and your location on Google Maps
Description:	This will display the location of each nearby event as well as your location on a Google Maps interface.
Primary Actor:	App
Preconditions:	This is the main page of the app. Once the user is logged in, this will be displayed
Postconditions:	All the locations will be displayed as pins on the map.
Main Success Scenario:	The location of the user is taken from the phone's gps and events are located using the Facebook API with the given location. Then a map of the nearby area is displayed with the locations shown by pins on the map and the user will be able to change the search radius.
Extensions:	If the location cannot be found, the user will be notified that their location cannot be found and that they need to turn on their location. If there are no events in their specified radius, they will be notified of that as well.
Frequency of Use:	This will be used every time the user accesses the app.
Status:	Backlog
Owner:	TBD
Priority:	Medium

ID:	**203**
Title:	Send notifications of upcoming events
Description:	The app will continually check the time and compare it with indicated events and when they are close enough, the app will send a notification.
Primary Actor:	App
Preconditions:	The app may or may not be open and the time before the event is lower than the threshold time.
Postconditions:	A notification is sent.
Main Success Scenario:	The app checks the time. It subtracts the time of the event from the current time. If the result is less than a certain time, a notification is sent.
Extensions:	If the time is greater than the threshold, nothing happens.
Frequency of Use:	1 notification will be sent per event.
Status:	Backlog
Owner:	TBD
Priority:	Low