Wikipedia-Pineapple Apple Pen Release Summary

Team members

Name and Student id	GitHub id	Number of story points that member was an author on.
Amanda Wai, 27518714 (Team Leader)	amawai	On This Day Notification (3/8 points) Recover Architecture (2/13 points) Total: 5 points
Sophia Quach, 40012754	sophiaquach16	On This Day Share Button (3/5 points) Burndown Chart (0.5 points) Recover Architecture (2/13 points) Total: 5.5 points
Jad Malek, 26345018	jadmalek	On This Day Notification Toggle (3/5 points) Recover Architecture (3/13 points) Total: 6 points
Chen Jie Lu, 27754388	SunXP	On This Day Notification (2/8 points) Recover Architecture (2/13 points) Total: 4 points
Maxim Nguyen, 27564171	mnhn329	On This Day Notification Toggle (2/5 points) Recover Architecture (2/13 points) Total: 4 points
Aman Bhandal, 27390858	abhandal	Mockups for Sprint 2 (1 point) On This Day Notification (3/8 points) Recover Architecture (2/13 points) Total: 6 points
Artem Mikhalitsin, 26349455	artemmikhalitsin	On This Day Share Button (2/5 points) Setup Development Environment (3 points) Ensure that Continuous Integration Runs All Tests (0.5 point) Total: 5.5 points

Mobile application summary

Wikipedia is one of the major pioneers in utilizing the internet prowess by granting free access to the largest virtual encyclopedia. The mobile application developed for Android aims to continue the vision by providing portable access to this vast knowledge base. The application includes exclusive features, such as *On This Day* articles, a timeline that shows historical events that happened during the present date and *Nearby,* a tool that displays articles relevant to the users surrounding. These exclusive features allow for a level of curiosity which enables our users to continually expand their knowledge anytime and anywhere.

Velocity

The commit for the first release has been tagged and can be accessed here.
Total: 8 stories, 36 points over 4 weeks
Iteration 1 (2 stories, 16 points)
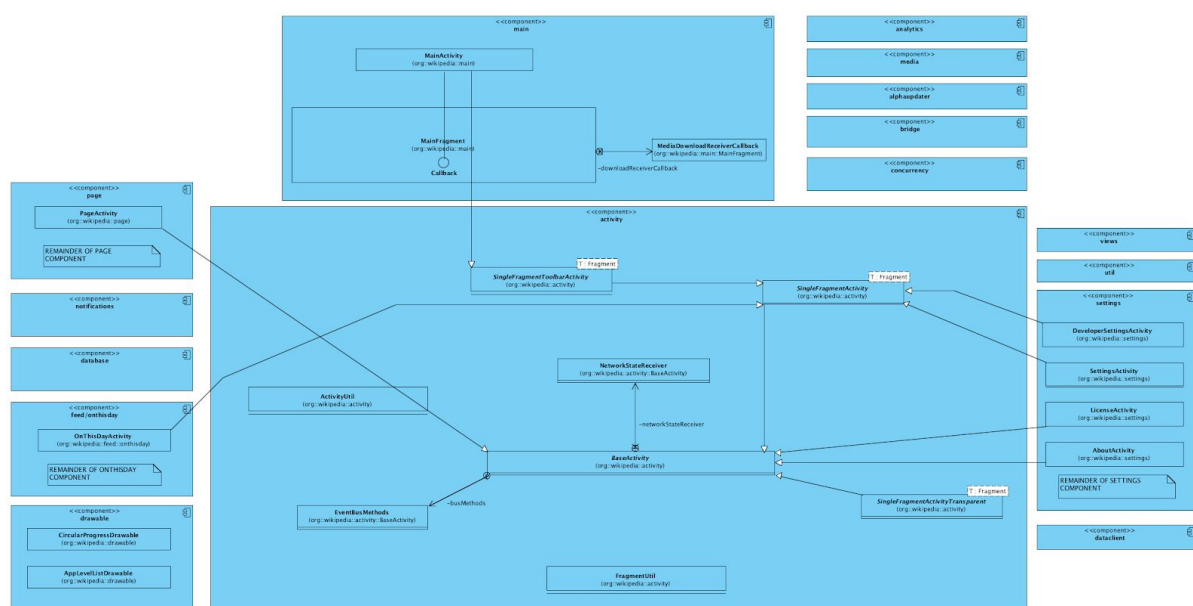Iteration 2, Release 1 (6 stories, 20 points)

Plan up to next release

Total: 8 stories, 45 points over 4 weeks and 5 days
Iteration 3 (4 stories, 22.5 points)
Iteration 4, Release 2 (4 stories, 1 epic account for over Iterations 3 and 4, 22.5 points)
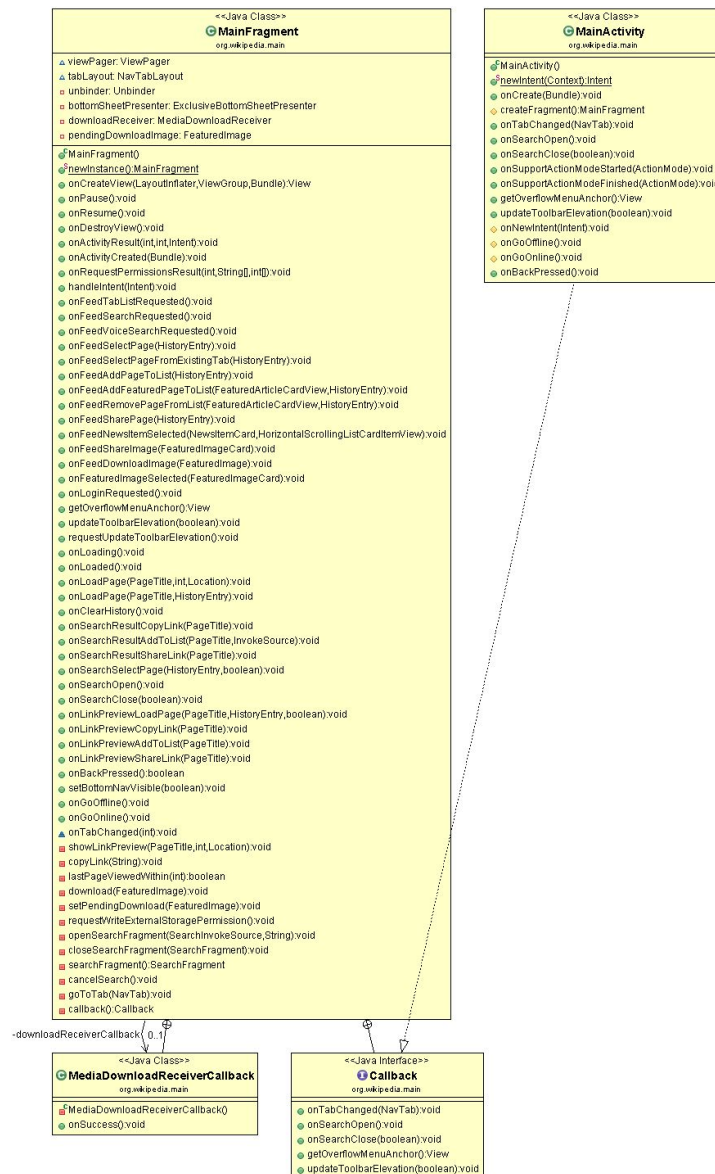
Overall architecture and class diagram

The diagrams recovered from the Wikipedia android application pertaining to the first release (and likely the next two releases) can be viewed here. Due to the size of the Wikipedia repository, the primary focus of the architecture analysis were the app\src\main\java\org\wikipedia subdirectories. For this reason, component and class diagrams were used to describe the subdirectories within the wikipedia subdirectory that were deemed the most important for features to be implemented in this and downstream releases. The following is a depiction of the overall components (considered significant for this team's purposes) within the wikipedia subdirectory and how they connect with one another.

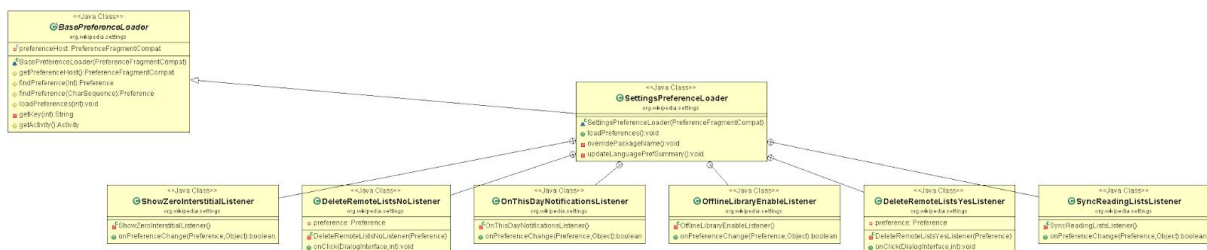Overall view of important components within wikipedia subdirectory

The following are the individual class diagrams for the classes that were modified during this release:

The activity component can be seen in the above component diagram, which has both controls for UI themes and the BaseActivity class depicted below.

<<Java Class>>
BaseActivity
org.wikipedia.activity

- BaseActivity()
- onCreate(Bundle):void
- onDestroy():void
- onResume():void
- onOptionsItemSelected(MenuItem):boolean
- onRequestPermissionsResult(int,String[],int[]):void
- setStatusBarColor(int):void
- setTheme():void
- setSharedElementTransitions():void
- onGoOffline():void
- onGoOnline():void
- onOfflineCompilationsFound():void
- onOfflineCompilationsError(Throwable):void
- searchOfflineCompilationsWithPermission(boolean):void
- onThisDayNotificationTransition():void
- searchOfflineCompilations(boolean):void
- requestStoragePermission():void
- showStoragePermissionSnackbar():void
- removeSplashBackground():void

-busMethods 0..1   -networkStateReceiver 0..1

<<Java Class>>
EventBusMethods
org.wikipedia.activity

- EventBusMethods()
- on(WikipediaZeroEnterEvent):void
- on(NetworkConnectEvent):void
- on(ThemeChangeEvent):void

<<Java Class>>
NetworkStateReceiver
org.wikipedia.activity

- NetworkStateReceiver()
- onReceive(Context,Intent):void

BaseActivity class diagram

The feed component which contains the onthisday subdirectory, the most essential piece for the sprint 2 features.

onthisday class diagram

The main component, which is critical for the launching of the application and control over its constituent activities:

main class diagram

Finally, the settings component, which contains the SettingsPreferenceLoader class (illustrated below) that enables the addition of entries to the settings.

SettingsPreferenceLoader class diagram

Infrastructure

The library, framework, tool, etc included in the app for this release is the following:

Name and link

[PowerMockito](#)

Reasons for using this framework

This framework was used in order to facilitate the mocking of classes with static and/or final methods. This was necessary in order to complete the testing of the notification functions. As the current version of Mockito used in the project does not currently allow for the mocking of final classes/methods, Powermockito was used.

Alternatives and why they do not work

An alternative to Powermockito is to use Mockito2, which has the added functionality of being able to mock final and static classes. However, updating to Mockito2 would render many of the current tests that were already written unusable due to incompatibilities between Mockito1 and Mockito2. It was decided that the overhead necessary to fix all of these tests was not worth the upgrade to Mockito2. Another alternative was to use Powermock with Easymock, as the syntax was extremely simple. However, the project already had mockito and other team members did not know about Easymock. For this reason, and to maintain consistency and simplicity, Powermock with EasyMock was not used.

Code

2 most important files that have been written:

File path with clickable GitHub link	Purpose
[android-wikipedia-390/app/src/main/java/org/wikipedia/feed/onthisday/OnThisDayNotifications.java](#)	Builds and sends the OnThisDay notification.
[android-wikipedia-390/blob/master/app/src/main/java/org/wikipedia/feed/onthisday/OnThisDayAlarmService.java](#)	Sets the alarm to activate the OnThisDay notification daily.

Testing and continuous integration

2 most important tests that have been written:

Test File path with clickable GitHub link	What is it testing
[/android-wikipedia-390/app/src/test/java/org/wikipedia/feed/onthisday/OTDNotificationTest.java](#)	This is testing the onReceive function of OnThisDayNotifications, ensuring that the notification displays the correct strings and that the right system service is called.
[app/src/test/java/org/wikipedia/feed/onthisday/OnThisDayCardTest.java](#)	The "Share OnThisDay article" string is formatted as expected.

Description of continuous integration environment

The link to the CI is: https://travis-ci.com/amawai/android-wikipedia-390. Travis, the continuous integration environment employed for the project, is free, hosted on GitHub and supports Java, the core language of this project. Whenever new commits are made and pushed to a repository, or a pull request is made, GitHub notifies Travis and Travis runs specific instructions written in the project's .travis.yml file. Once all the tests are complete, Travis notifies the developers about the build status and changes the tests' status in the pull request.