

Deliverable #1

Team Members: Hannah Posch, Alex Thropp, Daniel Baczmaga, Daniel Lee

1 Deliverable 1

1.0 Running project

To begin we first cloned the Glucosio Github Repository using the command, `git https://github.com/Glucosio`. After the project was cloned, we needed to install Android Studio and required android libraries through the terminal. Once Android Studio was installed, we were able to access the current tests created for Glucosio.

1.1 Existing Tests

Tests Completed within Android Studio:

❖ GlucosioApplicationTest

- 3 tests were included within this java file
 - ShouldNotClearLanguage_WhenAlreadyDone
 - ShouldClearLanguage_WhenSetFromHelloActivityAndNotFixedYet
 - ShouldSaveLanguageClearedToPreferences_WhenItIsDone
- All tests were passed and completed in 887 ms

❖ RobolectricTest

- Included a variety of test builds, however, if the option “All Tests” was chosen, the return stated that the test suite was empty
 - If each individual test was run, then the code worked correctly
- A1CCalculatorActivityTest
 - ReturnTrue_WhenKeyboardActionDone
 - Notify Presenter_WhenGlucoseValueChanged
 - ShouldAskPresenterToCheckUnit_When Created
 - ShouldBindViews_WhenCreated
 - ShouldSetMmolToMolAsUnit_WhenUserSettingsIsPercentage

- ReturnFalse_WhenKeyboardActionOther
- All tests were passed in 1m 7s
- AssistantAdapterTest
 - CallPresenter_WhenExportAsked
 - CallPresenter_WhenFeedbackAsked
 - CallPresenter_WhenA1CalculatorAsked
 - CallPresenter_WhenAddReadingAsked
 - CorrectlyBindRows
 - All tests passed 42s
- BackupActivityTest
 - ShouldDelegateToBackup_WhenDisconnectIsCalled
 - ShouldDelegateToBackup_WhenConnectIsCalled
 - ShouldDelegateToBackup_WhenActivityResultRecieved
 - ShouldInitBack_WhenCreated
 - All tests passed 41s
- HelloActivityTest
 - ShouldReportAnalytics_WhenCreated
 - ShouldInitLanguageSpinner_WhenCreated
 - ShouldSelectLanguage_WhenCreated
 - ShouldPassNLAsLocale_WhenNLSelected
 - ShouldBindView_WhenCreated
 - All tests passed 59s
- MainActivityTest
 - ShouldReportAnalytics_WhenCreated
 - Test passed 37s
- PreferencesActivityTest
 - ShouldReportAnalytics_WhenCreated
 - Test passed 38s
- AssistantAdapterTest

- CallPresenter_WhenExportAsked
- CallPresenter_WhenFeedbackAsked
- CallPresenter_WhenA1CCalculatorAsked
- CallPresenter_WhenAddReadingAsked
- CorrectlyBindRows
- All tests passed 39s
- BackupAdapterTest
 - ReturnViewWithSizeFormatted_WhenAsked
 - Test passed 30s
- MigrationTest
 - ProperlyMigrateUserFromVersion4
 - ProperlyMigrateGlucoseFromVersion4
 - ProperlyMigratePressureFromVersion4
 - ProperlyMigrateWeightFromVersion4
 - ProperlymigrateCholesterolFromVersion4
 - All tests passed 314 ms
- UserBuilderTest
 - Builds user account
 - All tests passed 1s
- AssistantFragmentTest
 - ShouldUnBindViews_WhenDestroyed
 - ShouldBindViews_WhenCreated
 - Tests passed 36s
- A1CCalculatorPresenterTest
 - 9 tests passed 433ms
- AssistantPresenterTest
 - CallFragment_WhenUserAskedExport
 - CallFragment_WhenUserAskedSupport
 - CallFragment_WhenUserAskedToAddReading

- CallFragment_WhenUserAskedToShowCalculator
 - All passed 92s
- ExternalViewPresenterTest
 - shouldLoadOpenSourceLicenses_WhenLicenseParameters
 - shouldThrowException_WhenNoParameters
 - shouldInvokeShowNoConnectionWarning_WhenNetworkIsNotConnected
 - All passed 28s
- HelloPresenterTest
 - 9 tests passed 28s
- OverviewPresenterTest
 - ShouldAddZerosBetweenReadings_WhenAsked
 - ShouldSortReadingsChronologically_WhenAsked
 - Tests passed 402ms
- ❖ AndroidPhoneAppTests
 - Required running emulators for test;
 - We are using an android emulator
 - HelloActivityTest
 - MainActivity Test

Test Speeds: We noticed that as the test were run across multiple systems, the speeds were inconsistent. We have determined that there is nothing significant to be worried about and that the difference in testing speeds is most likely a result of being run on a specific machine.

Organization: The project is organized in the following categories for which these tests were focused on.

- Activity
- Adapter
- Db
- Fragment
- Presenter

- Tools

1.2 Experiences

Overall, the project is fairly straightforward and easy to follow, Glucosio included plenty of documentation on how to set up and test/run their source code. The tests were thorough and easy to run, and we appreciated the pass/fail description for each subunit test. The ability to run all of the tests will need to be fixed for future runs, as that will allow testing to be more efficient, but we did not mind running the tests on their own as they are organized and easy to find in the directory. Of our group, Daniel L. owns a phone running an Android OS, so we hope to test on there.

Project:

The project is fairly robust and well organized. The developer made the code and terminology easy to follow with great documentation. We are personally excited to continue to work on this project and contribute to make the overall application better. We were able to run the app on an Android emulator, and hope to test on a physical Android device in the future.