

Name: Abhijeet Nikam

email: connect08nikam@gmail.com

IRC nickname: nikam08

Title: Active Mail - Implement API for composing mails and recreate the UI/UX for the mail composer.

Location: Coimbatore, India (UTC+05:30)

Motivation/Introduction:

Active Mail is a mail application which belongs to Kontact Touch produced by KDE-PIM Module and KDE Mobile Initiatives. It is the QtQuick based mobile version of KMail. It is designed to offer all the features of the desktop version of KMail client for touch screen devices, like smartphones, netbooks or tablets. KMail is a fully-featured email client that fits nicely into the KDE desktop. It has features such as support for IMAP, POP3, multiple accounts, mail filtering and sorting, PGP/GnuPG privacy, and inline attachments.

There is a list of recommendations to evaluate whether or not a specific application is suitable and eligible as a Plasma Active Application. There are currently issues which need to be fixed to follow the recommendations before Active Mail is truly called an Plasma Active Application.

Problems to be addressed:

1. QtWidget based dependency for many parts of the application like the email composer and viewer. This should be resolved for better visual interaction, touch friendly and interchangeable user interface.
2. Need of new API for the composing mails to implement the features of touch interfaces and creation of an all new composer for Active Mail.
3. According to recommendations: The app does not use pop-up windows, the user interface is contained as much as possible into one visual and interactive flow.

The Composer pops out in new window on clicking the compose mail icon. This has to be made inline with the main application window.

4. Present composer design is not up to mark for touch friendly experience. The UI needs to be reworked to make it more touch friendly.

These would be addressed by firstly writing an API for creating the new composer and to support touch-friendly interfaces with usage of QML to create the fluid interactions and UI. I have not done something like this before hence this will be a bit of challenge but a perfect opportunity to do and implement something new. It also presents a very exciting opportunity for me to learn so much during the project.

Project goals:

1. Create and implement an API for the composer and features for touch interfaces.
2. Using the API, implement the composer suitable for Active Mail's needs.
3. Work on the UI/UX part of the composer with the overall Active Mail structure and make it shine.

Future deliverables:

If time permits or after the time period, I would also love to fix the issue of picking up themes and styles as per the user's choice using the PlasmaComponents after we switch to the API approach. This fix would make the UI more awesome and attractive.

Implementation Details:

According to me, the project is clearly divided into three parts.

The creation of API will be the major part of the project. This will be done mostly in C++ and Qt. The present Active Mail already contains a composer with most of the functionalities. I will be going through the present code of the composer (most of the code lies in kdepim/messagecomposer) and write the new API suitable for QML imports in projects. This involves reusing the existing code, writing new ones where the codes are too old and difficult for independent breakdown into smaller modules or portion.

The API would be broken into following modules to perform the minimum requirements for a composer:

1. Create new text mail
2. Select to /CC/BCC
3. Reply to Sender/ALL
4. Reply to mailing list
4. Forward message
5. Attach files
6. Insert signature

Second comes the creation of the composer using the created API. This would be done as QML import using QDeclarativeExtensionPlugin.

I would be using QML and PlasmaComponents to create the composer features for fluid and touch friendly interfaces. I would try to make it inline my using the loader component on the new QML composer file. The composer would follow the swipe page format with advanced options menu bar being pulled from the side or top in a swipe. The final design will be based on the discussed mockup with experienced developer's and mentors for Active Mail.

Lastly, the finishing touches for UI/UX will be implemented on the composer and the Active Mail to make it more cool. This would again involve the usage of QML and PlasmaComponents for user interface. This is a minor part of the project compared to the first two ones.

While doing this project, I would like to spend some extra time working on documentation of the code, which will make the further development of this project much easier for new developers.

Before GSoC results are announced:

1. Set up the development environment and have all the prerequisites for the project development.
2. Do QML and Qt programming to create strong fundamentals in both of them.
3. Solve some easy/medium bugs to get more familiar with the source code.

Tentative Timeline:

Do you have other obligations from late May to early August?

I would be able to put 40 hours a week. My college semester exams start on 18th and end on 25th May. I would be able to put nearly 2-3 hours a day during this period. I would be compensating this by doing some early coding or working extra hours the following weeks. I have no other obligations during the summer's till early August and this project will be my primary goal. I am ready to put extra hours (>50) if required to achieve the set goals.

About Me:

I am a Computer Science and Engineering student at Amrita College of Engineering. I know C++,HTML and CSS. I know the basics of python and Qt, which I had used for the interface in my college project. I am presently sharpening my skills on QML and Qt. I have taken part in many open source meet ups and activities, and have taken part in spreading awareness about FOSS in my campus. I became more aware about KDE after I got selected for Hackfest under the KDE banner at IIT, Madras. I really like the community, the way new developers are guided and welcomed. I wish to become a core developer and a mentor too one day in this organisation. I am only focussing on this project and organisation by sending this one proposal for GSoC.

1. I have done some minor contributions - 20 coding style fixes for KDE PIM so I am familiar with the coding style and review process.
2. I know the basics of using Git Hub.
3. I am also presently working on the QML mockups for the composer in Active Mail.

