RAD: Project Platypus

Hidstrand, Fredrik
Keinestam, Johannes
Sjöqvist, Magnus
Thander, Fredrik

TDA367
Chalmers University of Technology
10 April 2011

Table of Contents

Requirements and Analysis Document for Project Platypus

Version: 2.0
Date: 2011-05-24

This version overrides all previous versions

1 Introduction

This section gives a brief overview of the project.

1.1 Purpose of application

The aim of the application is to provide the user with a way to quickly apply corrections and/or filters to large batches of images; either built-in filters, or customized ones that can be shared or made by one self. It's mainly geared towards photographers, who take large amounts of photos and might want to compress or apply effects to these.

1.2 General characteristics of application

PlatyPix will be a cross-platform desktop application: any machine that can run Java ought to be able to run it. The design will focus on ease-of-use, and follow a wizard-like design pattern.

1.3 Scope of application

The application will not support adding different filters to different images in the batch: all filters will be applied in the same way to all images. Support of vector images will not be included.

1.4 Objectives and success criteria of the project

- One should be able to apply filters to images chosen and save these onto the harddrive.

- A preliminary result of the added filters should be visible in a preview area.

- Custom filters should be able to be verified and added to the program's menu of available filters.

- Presets (a set of filters to be applied, along with needed values) should be savable by the user.

- One should be able to apply a saved preset to their image batch without any extra configuration of filters.

1.5 Definitions, acronyms and abbreviations

- Filter: any correction or effect added to an image with the purpose of modifying the look of it.

- Preset: a set of filters and corresponding values (e.g. scaling filter has values to describe scaling percentage or an absolute scaling size).

- Thumbnail: A small representation of a (usually) larger image.

2 Proposed application

In this section we propose an application.

2.1 Overview

PlatyPix will present a splash screen. When finished loading, the user will be prompted to add images to the batch. This screen will provide two choices: add filter or add preset. By choosing to add a filter, the user will be able to add filters to the batch from a list. If the user chooses to add a preset, they will choose from a list and the filters in that preset will be loaded.

When a filter is added, a view with a preview area and a filter settings area will be shown. The user can add several filters and navigate between added filters in the program. They finish the operation by saving the files to a specified path in a specified format.

2.2 Functional requirements

The user should be able to:

1. Load images to the batch with a file chooser dialog (and possibly drag-and-drop). Common bitmaps-based formats only: jpg, png, gif.
2. Add filters to operation from list of installed filters.
3. Change values for a filter, if needed. E.g. percentage to scale.
4. Save resulting images to disk in standard bitmap-based formats.
5. Save preset when editing is finished.
6. Load preset and apply to batch instead of/in conjuncture with adding filters manually.
7. Choose which preview image will be shown in the preview area when applying filters.
8. Import customized filter, which will be a .jar-file with:
    - Filter.class, the logic of the filter.
    - Additional libraries if needed.
9. Cancel operation. Will bring the user back to the start screen.

2.3 Non-functional requirements

2.3.1 Usability

The application will focus on ease-of-use, usability is therefor a concern. The way to use the program should be clear for anyone with even just an inkling of computer knowledge.

Minor tests during the development will be conducted with a novice computer user to guide us along the way. Localization of the program should also easy, but adding languages other than English will not be a primary priority.

2.3.2 Reliability

The application should reliably handle plugins (custom filters) and large images without faltering.

2.3.3 Performance

The time used to generate preview images on the fly while filters are being added or changed should be swift and not slow down the application (multi-threading).

Speed tests of the final step, rendering of resulting images, will be conducted in the end by applying several filters to large quantities of large images. This will be included in the documentation.

2.3.4 Supportability

Import of PNG, JPG, GIF, possibly TIFF should be supported. Exporting to these formats should also be supported, however with no format settings.

2.3.5 Implementation

The Java Runtime Environment (JRE) will be needed to run PlatyPix.

2.3.6 Verification

All included use cases will be subject to manual or automated tests (where possible). This will be included in the documentation.

2.3.7 Packaging and installation

PlatyPix will be delivered as a .zip-file containing:

1. Application code (executable .jar file)
2. Folders for resources, etc.
3. Scripts to start the application on different platforms.
4. README-file with startup instructions.

2.3.8 Legal

The license chosen for the application is the MIT License, giving the user total freedom to handle the software in whatever way they see fit.

2.4 Application models

Here we present an analysis of the domain and the functionality of the application

2.4.1 Scenarios

NA

2.4.2 Use case model

Use cases priority

1. Load images
2. Add filter
3. Change values on filter
4. Save images
5. Import filter
6. Apply filter on preview image
7. Save preset
8. Load preset

2.4.3 Static model

See APPENDIX 1.

2.4.4 Dynamic model

See APPENDIX 2.

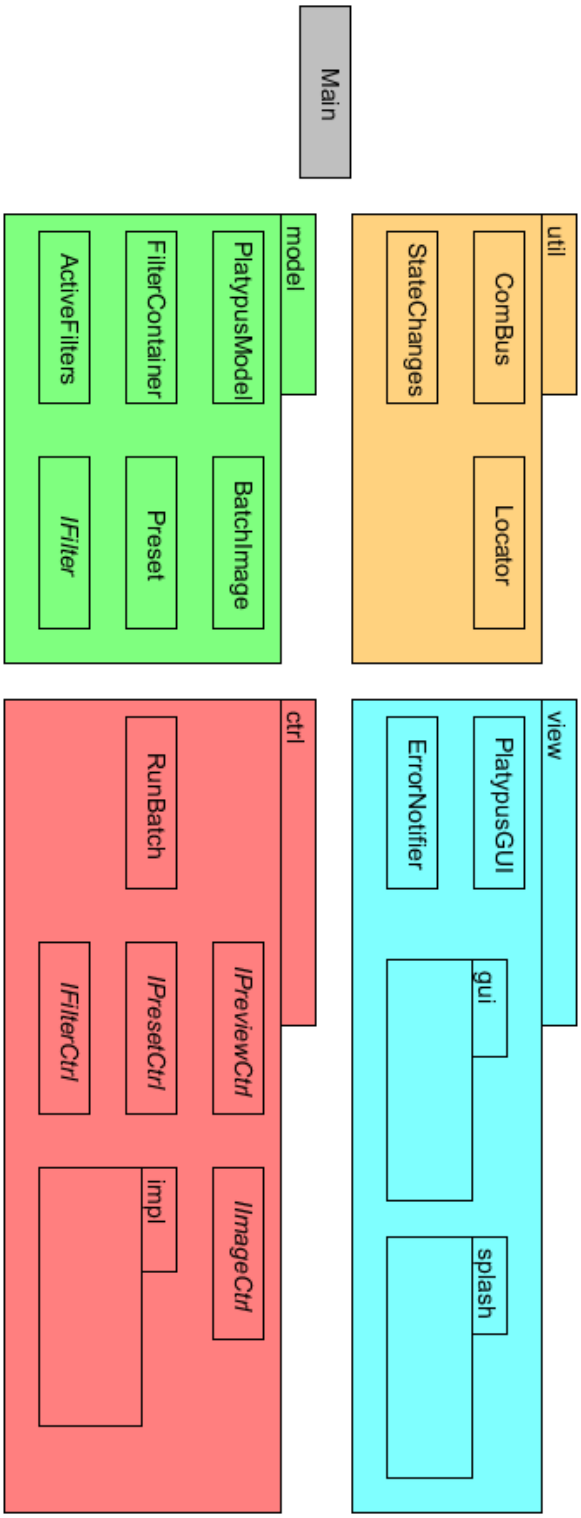
2.4.5 User interface

For sketches, see APPENDIX 3.

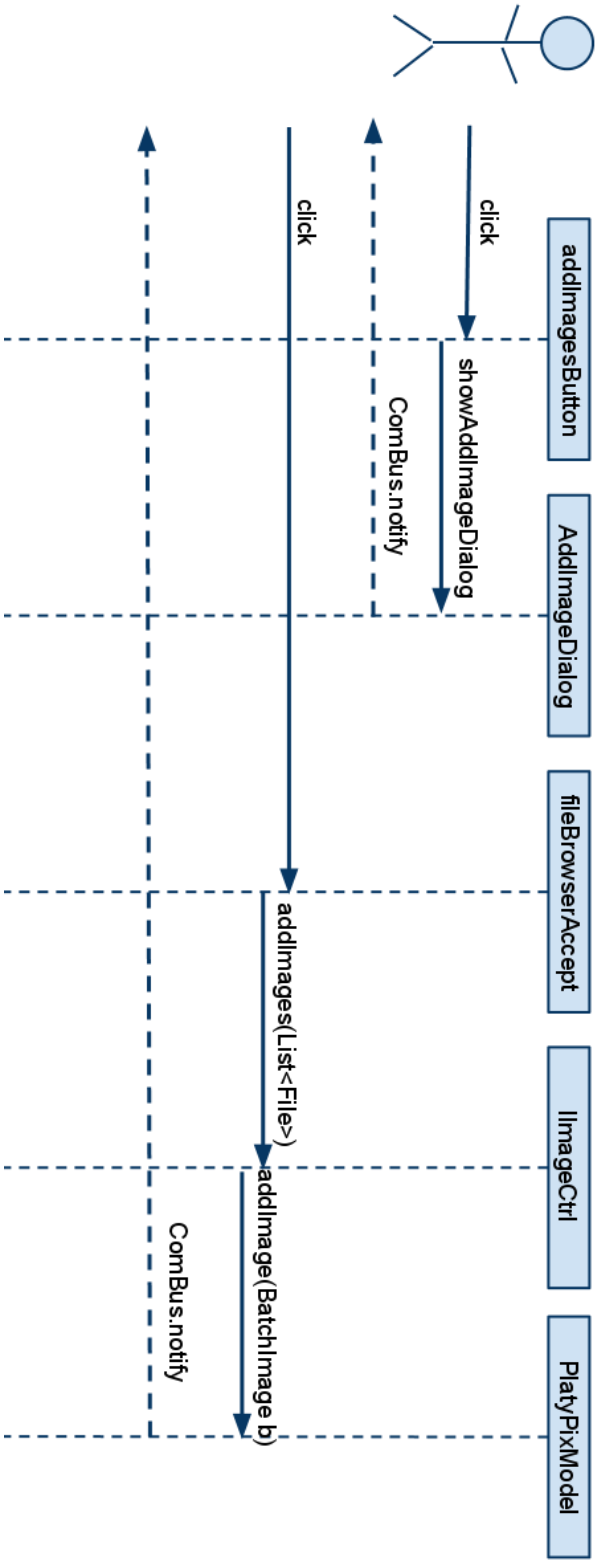
2.5 Possible future directions

- Proper optimization for handling large images
- Upload resulting images to online host
- Smartphone support

APPENDIX 1: Static model



APPENDIX 2: Dynamic model



APPENDIX 3: GUI sketches