

Engorged Wicket Pandoras

Account MAnager

EWP | Software Science | 14 December 2017

Courtney Appel

Jeff Bagley

Wyatt Featherly

Josh Tiefenthaler

Table of Contents

* Introduction: 2
* Purpose: 3
* System Overview
  + Basic System Design: 4
  + Architecture—MVC: 5
  + Functionality: 6
  + Non-Functional Requirements: 7
* System Requirements: 7
* System Security: 8
* Database Design: 8
* Our Team: 9
* Git Repository: 10

Introduction

*Account Manager* is a revolutionary new software developed by *Engorged Wicket Pandoras*. The purpose of *Account Manager* is to streamline the out-of-date and inefficient methods used by the Computer Science department at the University of Montana to record and modify transactions within the department.

Previously, a simple transaction (such as a deposit by a student for camp fees) required an entire army to process. Multiple staff members would have to email, edit spreadsheets, calculate fees, submit to UM’s data management system, and even physically run all over campus (just for one student’s deposit!).

*Account Manager* aims to help simplify and optimize this process while cutting down on the headaches endured by the departmental staff.

Purpose

*Account Manager’s* first goal in order to improve upon the system in place was to eliminate the need to enter information multiple times. In order to create a new type of account the admin need only select ‘new account’ and enter the relevant information. In order to enter a new a transaction the admin need only select ‘new deposit’ or ‘new withdrawal’ and enter the relevant information. This info is saved and can be accessed at any time.

The second goal to expand the system was to eliminate the need for the user to calculate fees. The software automatically subtracts university/credit card fees from deposits and includes a benefits calculator in order to calculate the amount taken from wages that go toward benefits.

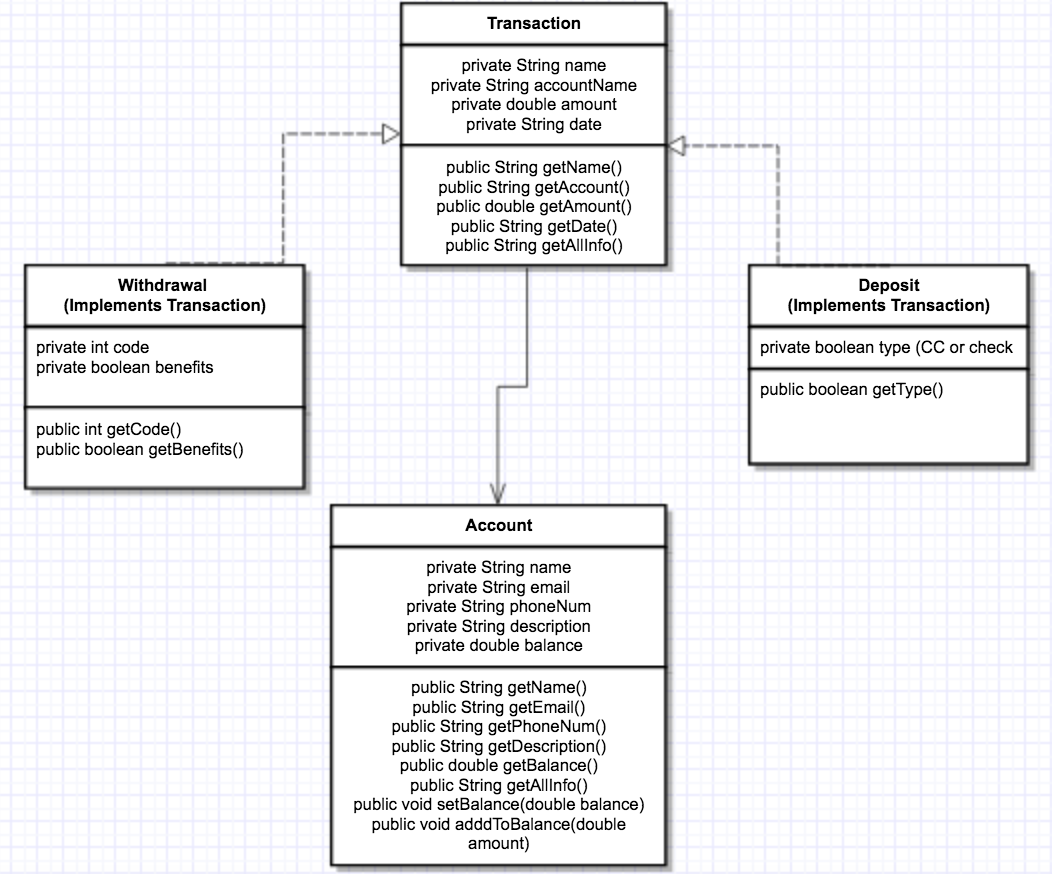
The third goal was to increase control for the user. The admin can view transaction reports for any individual user or the entire department with the click of a button. The admin can also edit any information and can delete accounts/transactions without it adversely affecting the reports.

The fourth goal was to increase security for the user. Login requires a username and password and without that information there is no way to access the potentially confidential information included in the software.

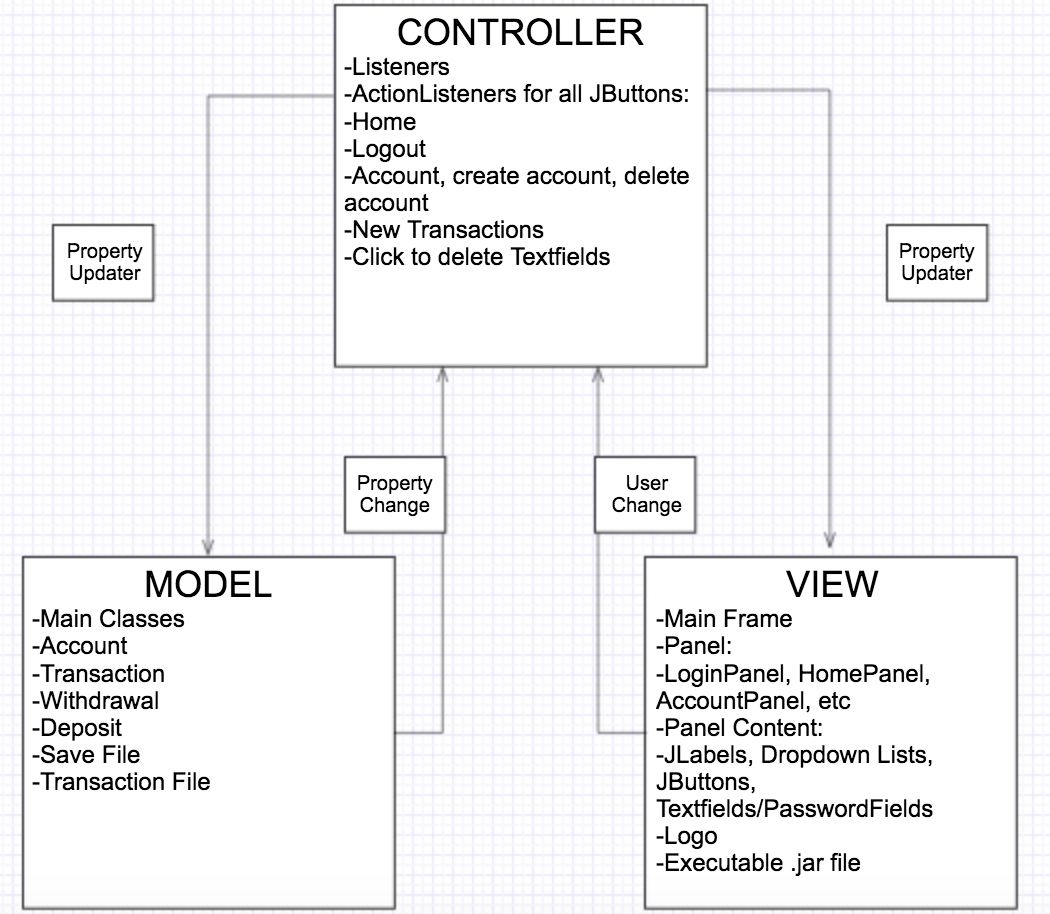
The final goal was to make the experience more user friendly and easy to use. All functions and information are easy to locate and the user can always return to the home screen or log out with a single click.

System Overview

Basic System Design



Architecture—MVC



Functionality

Functional features as of release 1 include:

* Admin login (username: csadmin, password: csci323)
* Logo and team name is always at bottom of screen
* Software Name is always at top of screen
* Ability to log out from any screen
* Ability to go to home view from any screen (once logged in)
* Ability to close window from any screen
* Benefits Calculator is always at top of screen
* All information is saved upon close
* Home view includes Accounts, New Withdrawal, and New Deposit
* New Account requires name, description, phone, email, all fields must be filled, can’t have multiple identical accounts
* Admin can edit account details and delete accounts (except for the master account)
* Admin can view transactions for each account (including master) and can delete transactions
* All transactions require date (auto-populates), description, amount, name, and all fields must be filled
* Deposits apply appropriate fees (8% university fee and 4% cc fee)
* Withdrawals have codes available, user can add new codes.

Non-Functional Requirements

Because *Account Manager* is still in its early stages, it has few non-functional requirements.

* A*ccount Manager* REQUIRES Java installed on the user’s machine. Java does not need to be fully up to date but versions prior to Java 8 may have small eccentricities.
* *Account Manager* is ran from an executable .jar file which is very small in size, about 50 kb.
* The user is NOT required to be connected to the internet at this point in time.
* *Account Manager* runs on OSX and Windows with slight modifications depending on your version. A Linux version is in the works.
* No major hardware is required and *Account Manager* should run smoothly on any modern computer.

System Requirements

As noted above, any modern computer running OSX or Windows should run *Account Manager* just fine as long as Java is installed.

System Security

As of release 1, *Account Manager* only allows for administrative login. The login username is *csadmin* and the password is *csci323*. These credentials are not case sensitive.

If the username or password is entered incorrectly the login screen will prompt the user to enter new credentials.

The password is set as a PasswordField, so no characters are visible and are instead replaced with nondescript characters.

Database Design

As of release 1, a database has not been fully implemented. Rather, all information is stored in 2 text files which are located in the project folder.

Our Team

All group members contributed significantly to the main panel which was the bulk of the project. However, each member had their own smaller responsibilities.

|  |  |
| --- | --- |
| ../assign2/cna.gif | Courtney Appel: The creative mind. Designed our logo as well as MVC diagram and slideshow used in our sprints. |
| ../assign2/jeff.gif | Jeff Bagley: Scrum Master. Presented product at frequent sprints and designed System Document. |
| ../assign2/wyatt.gif | Wyatt Featherly: Cosmetics and aesthetics. Designed buttons/background image. Formatted panel to look nice. |
| ../assign2/josh.gif | Josh Tiefenthaler: Functionality. Designed and implemented functions to make the program actually work. |

Git Repository

Software is located at <https://github.com/courtneyappel/EWP>.

Note: Executable .jar file is titled ‘AccountManagerRelease1.jar’.