# Assignment 3 - SDSUI

Regina Weeks - Chase Denecke - Brice Ng - Tyler Farnham - Qibang Liu

February 5, 2018

1 UI Prototypes

1. HOME SCREEN: This is the login and sign up screen. The user simply enters their email and password and clicks sign in or sign up depending on if they are already user or not.

2. SETTINGS: The settings page is where the user can set the amount of money that will be donated and to what charity. They can also set the amount of time they would like to remain focused for. They can add apps to their list of time wasting apps by pressing the "ADD APPS" button.

3. SUMMARY: The summary screen provides the user with their list of what they consider to be time wasting apps. Pressing the button begins their flagellation session.

4. FOCUS OR REPENT: This screen indicates that the Flagellant is currently running and if the user opens up any time wasting apps within the designated amount of time, their money will be donated. For testing purposes, the user can press the "REGRET" button and have their money refunded to them.

FLAGELLANT
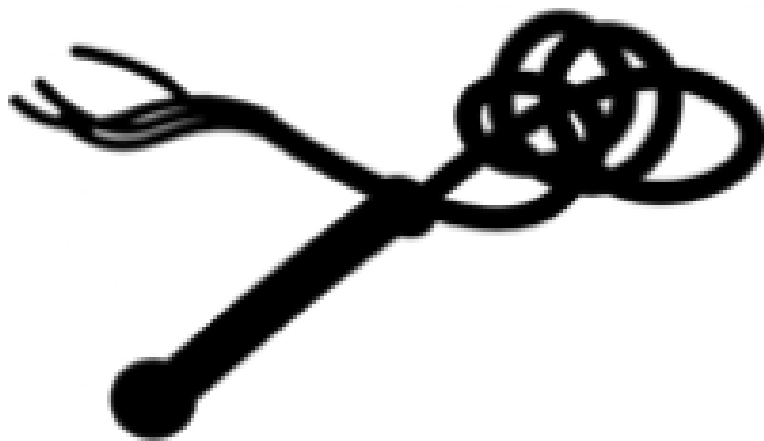
enter username (email)

enter password

SIGN IN   SIGN UP

SETTINGS

≡

AMOUNT TO DONATE $20

CHANGE CHARITY   Flagellant Developers ▾

SET FOCUS TIME   1 HOUR

ADD APPS

SUMMARY

☰

YOUR TIME WASTING APPS

Reddit
Facebook
Tinder
▲
▼

PRESS BUTTON TO BEGIN
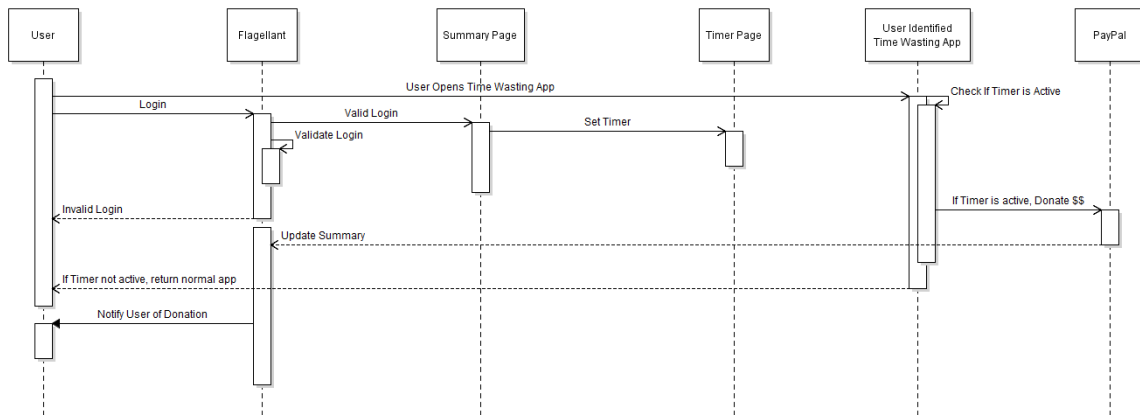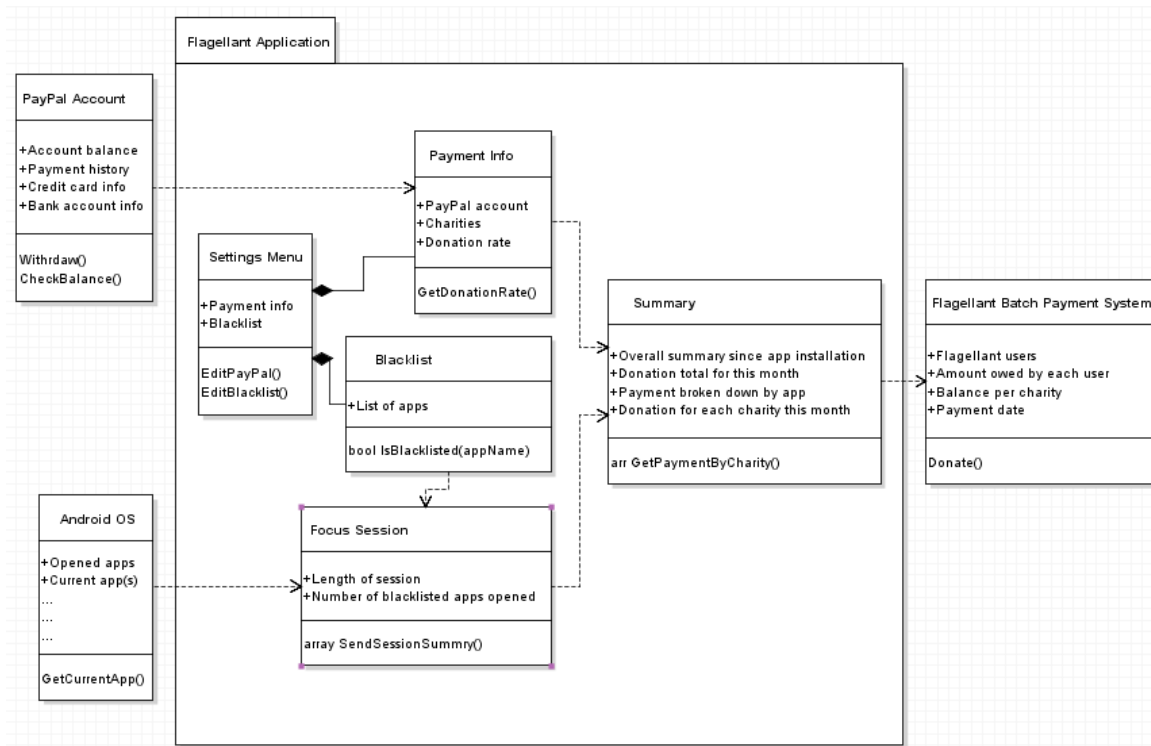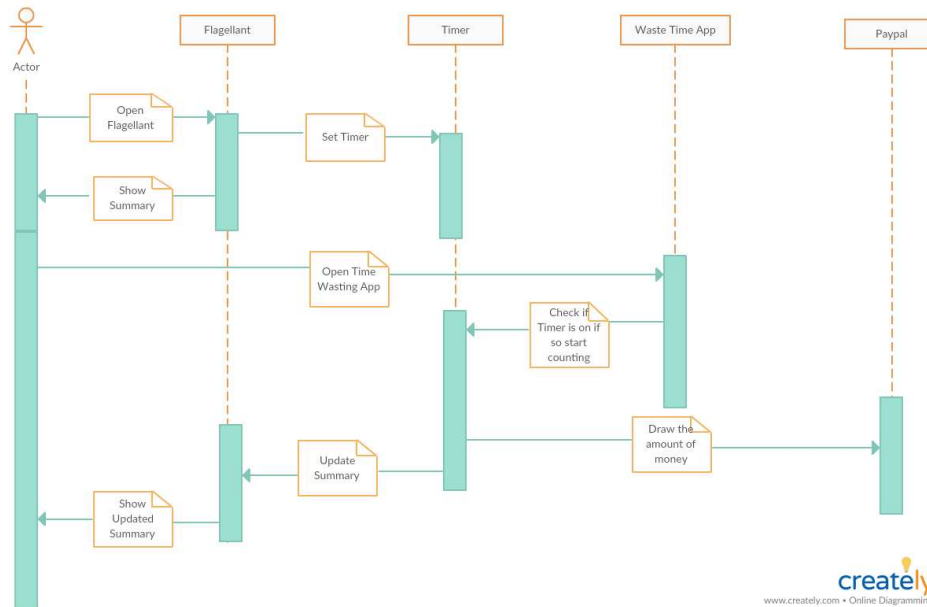FOCUS PERIOD

BEGIN FLAGELLATING

FOCUS OR REPENT

FLAGELLANT IS RUNNING
USE OF TIME WASTING APPS
WILL RESULT IN YOUR MONEY
BEING DONATED TO CHARITY

59:01
REMAINING

REGRET

Flagellant Application

PayPal Account

+Account balance
+Payment history
+Credit card info
+Bank account info

Withrdaw()
CheckBalance()

Payment Info

+PayPal account
+Charities
+Donation rate

GetDonationRate()

Settings Menu

+Payment info
+Blacklist

EditPayPal()
EditBlacklist()

Blacklist

+List of apps

bool IsBlacklisted(appName)

Summary

+Overall summary since app installation
+Donation total for this month
+Payment broken down by app
+Donation for each charity this month

arr GetPaymentByCharity()

Flagellant Batch Payment System

+Flagellant users
+Amount owed by each user
+Balance per charity
+Payment date

Donate()

Android OS

+Opened apps
+Current app(s)
...
...
...

GetCurrentApp()

Focus Session

+Length of session
+Number of blacklisted apps opened

array SendSessionSummry()

Sequence diagram actors: User, Flagellant, Summary Page, Timer Page, User Identified Time Wasting App, PayPal

- User Opens Time Wasting App
- Login
- Valid Login
- Validate Login
- Set Timer
- Check If Timer is Active
- Invalid Login
- If Timer is active, Donate $$
- Update Summary
- If Timer not active, return normal app
- Notify User of Donation

Our sequence Diagrams for our three use cases are near identical, as the only thing that changes is the app that is being opened. The Use cases "Motivation for cramming an assignment" and "not ignoring your friends" are identical aside from the app opened as a distraction, so only one sequence diagram was made. Flagellant has one purpose; to dissuade app use, and it has limited setting and functionality otherwise.

2 Meeting Report

This week we have accomplished several things. First of all, we completed Assignment 3, with more attention to the sub-category requirements than last week. In addition, each of us completed at least the first module of SoloLearn's Java tutorial. A couple of us completed up to module 3. Tyler set up a couple of UI screens for our app using Android studio, complete with animations to show and hide screens. Alea learned how to connect to an API and get data from the API, as well as how to separate UI tasks from background tasks. This next week we plan to meet on Friday to start creating UI screens and possibly get started on linking the PayPal API to our app. Our stretch goal is to read through some notes Alea shared with us in our Google Doc that detail the basics of Android app development.

For Assignment 3, work was completed as follows. Alea did the UI mocks for all 3 use cases. We ended up with the same UI screens for all 3 use cases so we submitted one set of UI mocks for all 3 use cases. Austin created a sequence diagram for the use case in which a user buys too much stuff online and wants to stop shopping so much. Brice created a sequence diagram for use cases 1 and 2 (Motivation while cramming an assignment and Not ignoring your friends). Chase created the UML class diagram and wrote the Meeting Report. Tyler assembled everyone's contributions and compiled them into Latex.

Our customer was indeed able to meet with us. In fact, our customers are part of the development team. They form an embedded unit within our development team, facilitating rapid communication, quick iteration cycles, and team synergy.