System and Software Architecture Description (SSAD)

PicShare

Team 02

Team Members
Sultan Alsarra
Adil cem Albayrak
Julapat Julnual
Charles Reitz
Mohammad Almunea
Aref Shafaeibejestan
Andrea Brown
Travis Weaver
Dennis Evans

4/25/16

Version History

Date	Author	Version	Changes made	Rationale
10/18/15	Adil & Aref	0.9	• Added Introduction and System Analysis	• Initial draft for the FC package.
10/25/15	Adil & Aref	1.0	• Added admin feature	• Completed for FC Package
11/29/15	Adil, Aref & Mohammad	2.0	• Analysis Sequence Diagram Added • Complete Process Descriptions • Analysis Class Diagram Added	• Complete Draft for DCR
12/07/15	Aref	2.1	• ERD Added.	• Complete for DC Package
2/11/16	Aref, Adil & Mohammad	2.2	• Add Architectural styles, patterns and frameworks section, update class diagram and ER diagram.	• Complete overall Architecture
3/2/16	Sultan, Aref & Mohammad	2.3	• Update Use case descriptions, ERD and class Diagram	• Revise for Design Review
4/25/16	Aref	2.4	• Update the sequence diagrams	• Complete SSAD for as built package.

Table of Contents

Table of Tables

Table of Figures

1. Introduction

1.1 Purpose of the SSAD

The purpose of the SSAD is demonstrating the details about the system architecture, software and hardware parts that will be used in the project. The report presents the key properties of the system by analyzing the system context diagram and showing the use cases.

1.2 Status of the SSAD

This is Version 2.4 of System and Software Architecture Description.

2. System Analysis

2.1 System Analysis Overview

The primary purpose of the PicShare application is to share pictures easier than it is today. With PicShare, users can share pictures in three different ways:

1. Post pictures to the “near-by” location. So everyone in the near by can view these pictures.
2. Post pictures to public events by using a hash tag. In this way, pictures that are related to a certain public event will be put together.
3. Post pictures to a private event. Private events can be set up that only people with password can access them.

2.1.1 System Context

Figure 1: System Context Diagram

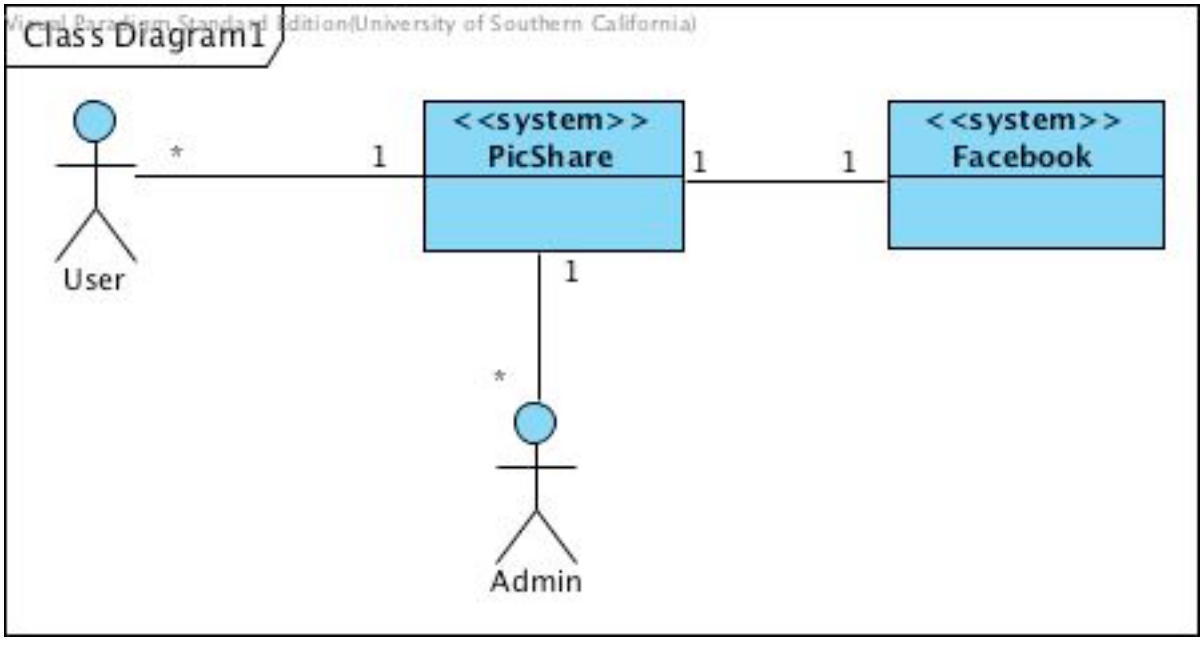


Table 1: Actors Summary

Actor	Description	Responsibilities
User	User of the system.	Sharing picture. Creating public and private event.

Actor	Description	Responsibilities
Admin	Monitors the system	Viewing pictures and events and moderating them to control the whole system.

2.1.2 Artifacts & Information

Figure 2: Artifacts and Information Diagram

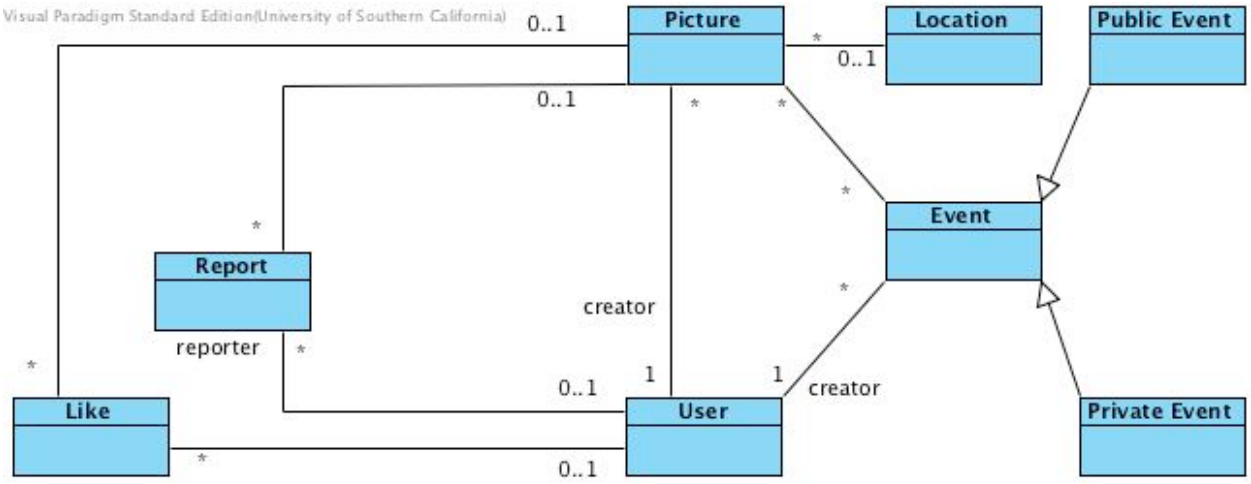


Table 2: Artifacts and Information Summary

Artifact	Purpose
Picture	Contains all information about picture, including the picture itself and id indicating the picture.
Location	Contains longitude and latitude for each location.
User	Contains username (email) and password of the user.
Report	Contains information about reporting. That is the user id, picture id and the reason of the report.
Like	Indicates which user has liked which picture.
Event	Contains event name (hashtag) for each event.
Private Event	Indicates the private events and contains password alongside the hashtag.
Public Event	Indicates the public events that will be created by a hashtag.

2.1.3 Behavior

Below you can see the process diagram (use case diagram). In the next sections we are going to describe some of the use cases.

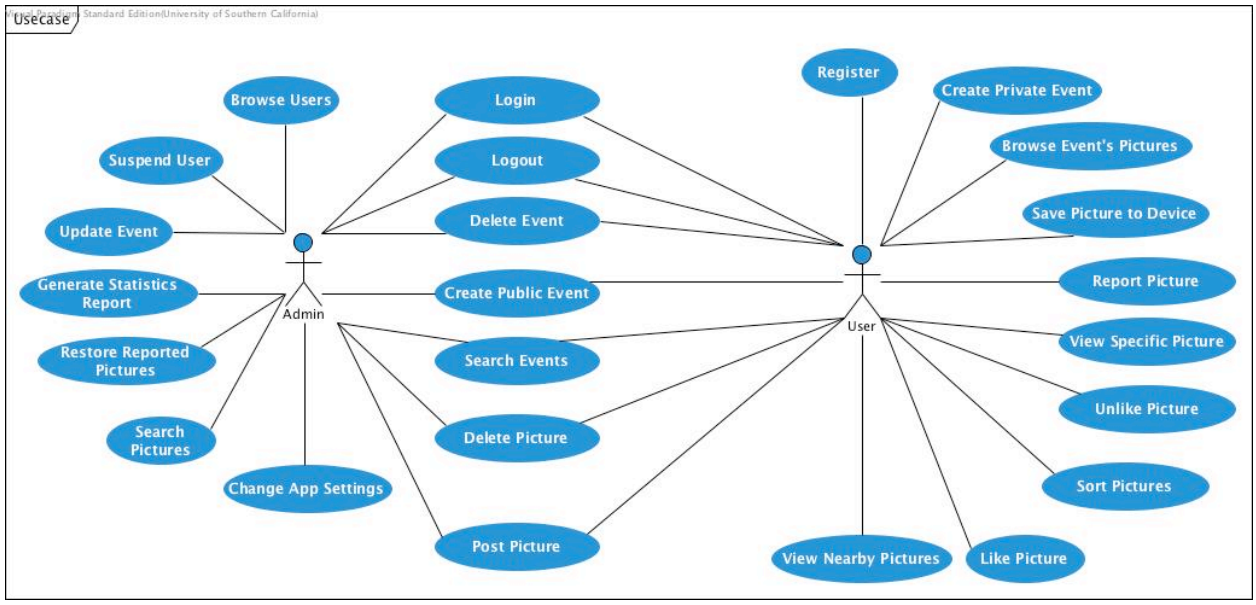


Figure 3: Process Diagram,

2.1.3.1.1 Process Post Picture

Table 3: Process Description: Post Picture

Identifier	UC-7: Post Picture
Purpose	Allow a user or admin to add a picture to location or hashtag.
Requirements	WC_3579, WC_3619
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	User adds a picture to the system.

Table 4: Typical Course of Action- Post Picture: Taking Picture

Seq#	Actor's Action	System's Response
1	User/admin clicks on camera icon button.	
2		System opens up the camera application.

3	User takes a photo.	
6		System shows the picture and also option to post to nearby or to an event
7	User chooses "Post to Nearby" and clicks submit.	
8		System retrieve user's location and posts the picture in that location.

Table 5: Alternate Course of Action- Post Picture: Uploading Picture

Seq#	Actor's Action	System's Response
1-2	Refer to the typical course of action step 1-2	
3	User/admin selects "LIBRARY" option.	
4		System opens up the gallery application.
5	User/admin selects a photo and clicks "√" marks	
6		System shows the picture and a text box for user to input caption and Select an Event or Post to Nearby.
7	User/admin Selects an Event and enters caption and clicks submit.	
8		System stores the picture in the event selected with caption.

Table 6: Alternate Course of Action- Post Picture: Hashtag Option

Seq#	Actor's Action	System's Response
1	Refer to the typical course of action step 1-6	
2	User/admin chooses Select an Event option.	
3		System shows a text box for user to input the name of event name.
4	User/admin enters the event name and caption and clicks submit on it.	
		System stores the picture with the hashtag and caption.

2.1.3.1.2 Process Delete Picture

Table 7: Process Description: Delete Picture

Identifier	UC-8: Delete Picture
Purpose	Allow a user to delete his/her pictures.
Requirements	WC_3591
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	User removes a picture that belongs to him/her from the system.

Table 8: Typical Course of Action- Delete Picture: Confirm

Seq#	Actor's Action	System's Response
1	User clicks on Delete button.	
2		System shows a warning to the user to confirm this action.
3	User selects "Delete" option.	
4		System closes the warning and deletes the picture.

Table 9: Alternate Course of Action- Delete Picture: Cancel

Seq#	Actor's Action	System's Response
1-2	Refer to the typical course of action step 1-2	
3	User selects "Cancel" option.	
4		System closes the warning and the picture remains.

2.1.3.1.3 Process View Specific Picture

Table 10: Process Description: View Specific Picture

Identifier	UC-9: View Specific Picture
Purpose	Allow a user to view picture details.
Requirements	WC_3637
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	Details of a specific picture is shown.

Table 11: Typical Course of Action- View Specific Picture

Seq#	Actor's Action	System's Response
1	User click on picture to view specific picture.	
2		System retrieves the picture details (picture, location or hashtag, number of likes, a way to like/unlike picture).

2.1.3.1.4 Process Like Picture

Table 12: Process Description: Like Picture

Identifier	UC-10: Like Picture
Purpose	Allow a user to like a picture.
Requirements	WC_3751
Development Risks	None
Pre-conditions	• User is logged in the system. • User has not liked the picture before.
Post-conditions	User likes the specific picture.

Table 13: Typical Course of Action- Like Picture

Seq#	Actor's Action	System's Response
1	User click on heart icon for a specific picture.	
2		System stores the like information and new total number of likes is shown to the user.

2.1.3.1.5 Process Unlike Picture

Table 14: Process Description: Unlike Picture

Identifier	UC-11: Unlike Picture
Purpose	Allow a user to unlike a picture that he/she has liked before.
Requirements	WC_3751
Development Risks	None
Pre-conditions	• User is logged in the system. • User has liked the picture before.
Post-conditions	User unlikes the specific picture.

Table 15: Typical Course of Action- Unlike Picture

Seq#	Actor's Action	System's Response
1	User click heart icon to unlike a specific picture.	
2		System removes user liking that picture and new total number of likes is shown to the user.

2.1.3.1.6 Process Report Picture

Table 16: Process Description: Report Picture

Identifier	UC-12: Report Picture
Purpose	Allow a user to report pictures.
Requirements	WC_3599
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	User reports the specific picture.

Table 17: Typical Course of Action- Report Picture: Submit

Seq#	Actor's Action	System's Response
1	User click three button icon for a picture and clicks on report	
2		System stores the report.

Table 18: Alternate Course of Action- Report Picture: Cancel

Seq#	Actor's Action	System's Response
1-2	Refer to typical course of actions step 1-2	
3	User clicks on "cancel" button.	
4		System closes the options page.

2.1.3.1.7 Process Save Picture to Device

Table 19: Process Description: Save Picture to Device

Identifier	UC-13: Save Picture to Device
Purpose	Allow a user to save a picture to device.
Requirements	WC_3623
Development Risks	None
Pre-conditions	User is logged in the system.

Post-conditions	User stores the specific picture in her/his own device.

Table 20: Typical Course of Action- Save Picture to Device

Seq#	Actor's Action	System's Response
1	User clicks three button icon to selects option to download a specific picture.	
2		System saves the picture to the user's device.

2.1.3.1.8 Process Browse Event's Pictures

Table 21: Process Description: Browse Event's Pictures

Identifier	UC-14: Browse Event Pictures
Purpose	Allow a user to view pictures in an event.
Requirements	WC_3637
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	User views the pictures in a public/private event.

Table 22: Typical Course of Action- View Event Pictures

Seq#	Actor's Action	System's Response
1	User clicks hashtag icon and enters event name into text input.	
2		System shows search result
3	User clicks the desired event name	
4		System shows the pictures related with the event to the user.

2.1.3.1.9 Process Sort Pictures

Table 23: Process Description: Sort Pictures

Identifier	UC-15: Sort By Like or Date or location
Purpose	Allow a user to sort pictures by popularity, date or location.
Requirements	WC_3637
Development Risks	None

Pre-conditions	• User is logged in the system. • A list of pictures is shown to user.
Post-conditions	User sorts the picture by popularity, date or location.

Table 24: Typical Course of Action- Sort By Like, Date or Location

Seq#	Actor's Action	System's Response
1	User clicks filter icon on top right.	
2		System displays options to sort pictures by popularity, date or location.
3	User selects the options they want	
4		System sorts the pictures in according to the option user chose.

2.1.3.1.10 Process Create Private Event

Table 25: Process Description: Create Private Event

Identifier	UC-4: Process Create Private Event
Purpose	Allow a user to create a private event for invited people.
Requirements	WC_3621
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	User creates a private event that only invited people can attend.

Table 26: Typical Course of Action - Create Private Event: With a Picture: Event does not exist

Seq#	Actor's Action	System's Response
1	User takes a picture and posts it in a private event.	
2		System shows the private event name and password fields to user to fill.
3	User fills the private event name field.	
4		System checks presence of the event and warns the user.
5	User fills the password field and	

	clicks “submit” button.	
6		System creates the private event.

Table 27: Alternate Course of Action- Create Private Event: With a Picture: Event exists

Seq#	Actor’s Action	System’s Response
1-4	Refer to typical course of actions step 1-4	
5	User changes the name, fills the password field and clicks the “submit” button.	
6		System creates the private event.

Table 28: Typical Course of Action - Create Private Event: Without a picture: Event does not exist

Seq#	Actor’s Action	System’s Response
1	User clicks the “Create New Event” button.	
2		System shows the private event option, event name and password fields to user to fill.
3	User fills the private event name field.	
4		System checks presence of the event and warns the user.
5	User fills the password field and clicks “Create Event” button.	
6		System creates the private event.

Table 29: Alternate Course of Action- Create Private Event: Without a picture: Event exists

Seq#	Actor’s Action	System’s Response
1-4	Refer to typical course of actions step 1-4	
5	User changes the name, fills the password field and clicks the “Create Event” button.	
6		System creates the private event.

2.1.3.1.11 Process Delete Event

Table 30: Process Description: Delete Event

Identifier	UC-5: Process Delete Event
Purpose	Allow admin and a user to delete a private event.
Requirements	WC_3603
Development Risks	None
Pre-conditions	• User is logged in the system. • User is creator of the private event or admin is deleting an event.
Post-conditions	User deletes a private event with all associated pictures.

Table 31: Typical Course of Action - Delete Event: Submit

Seq#	Actor's Action	System's Response
1	User/admin asks system to delete an event.	
2		System shows a warning to the user to confirm this action.
3	User/admin selects "Ok" option.	
4		System closes the warning and deletes the private event with all associated pictures.

Table 32: Alternate Course of Action- Delete Event: Cancel

Seq#	Actor's Action	System's Response
1-2	Refer to typical course of actions step 1-2	
3	User/admin clicks on "cancel" button.	
4		System closes the warning and the private event remains.

2.1.3.1.12 Process Search Events

Table 33: Process Description: Search Events

Identifier	UC-6: Process Search Events.
Purpose	Allow a user to search the presence of an event.
Requirements	WC_3727
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	User searches the presence of an event in the system.

Table 34: Typical Course of Action - Search Events

Seq#	Actor's Action	System's Response
1	User enters the name of event in the search box.	
2		System searches the presence of that event and shows the results in the same page.

2.1.3.1.13 Process Create Pubic Events

Table 35: Process Description: Create Public Events

Identifier	UC-17: Process Create Public Events.
Purpose	Allow a user to create public events.
Requirements	WC_3621
Development Risks	None
Pre-conditions	User is logged in the system.
Post-conditions	User creates a public event that everyone can attend and post pictures.

Table 36: Typical Course of Action - Create Public Event: With a Picture: Event does not exist

Seq#	Actor's Action	System's Response
1	User takes a picture and posts it in a public event.	
2		System shows the public event name field to user to fill.
3	User fills the public event name field.	
4		System checks presence of the event and warns the user.
5	User makes sure that there is no event created before and clicks "submit" button.	
6		System creates the public event.

Table 37: Alternate Course of Action- Create Public Event: With a Picture: Event exists

Seq#	Actor's Action	System's Response
1-4	Refer to typical course of actions step 1-4	
5	User changes the name and clicks the "submit" button.	
6		System creates the public event.

Table 38: Typical Course of Action - Create Public Event: Without a Picture: Event does not exist

Seq#	Actor's Action	System's Response
1	User clicks the "Create New Event" button.	
2		System shows the public event name field to user to fill.
3	User fills the event name field and clicks "Create Event" button	
4		System checks presence of the event and if there is no conflict, it creates the event. If event exists, it gives user warning

Table 39: Alternate Course of Action- Create Public Event: Without a Picture: Event exists

Seq#	Actor's Action	System's Response
1-4	Refer to typical course of actions step 1-4	
5	User changes the name and clicks the "Create Event" button.	
6		System creates the public event.

2.1.3.1.14 Process Browse Users

Table 40: Process Description: Browse Users

Identifier	UC-18: Browse Users
Purpose	Allow admin to browse users.
Requirements	WC_3765
Development Risks	None
Pre-conditions	Admin is logged in the system.
Post-conditions	A list of users is shown to admin and he can search users.

Table 41: Typical Course of Action - Browse Users

Seq#	Actor's Action	System's Response
1	Admin asks system to display a list of users.	
2		System shows a list of users to admin, alongside a form that admin can use to filter the list.
3	Admin enters information to filter the list.	
4		System displays the new list of users based on the filter form.

2.1.3.1.15 Process Suspend User

Table 42: Process Description: Suspend User

Identifier	UC-19: Suspend User
Purpose	Allow admin to suspend a user
Requirements	WC_3766
Development Risks	None
Pre-conditions	Admin is logged in the system.
Post-conditions	A user is suspended

Table 43: Typical Course of Action - Suspend User: Confirm

Seq#	Actor's Action	System's Response
1	Admin asks system to suspend a user.	
2		System asks admin if he/she is sure about the action.
3	Admin clicks on "ok" button	
4		System suspends the user.

Table 44: Typical Course of Action - Suspend User: Cancel

Seq#	Actor's Action	System's Response
1	Refer to typical course of actions step 1-2	
3	Admin clicks on "cancel" button	
4		System cancels the suspending action.

2.1.3.1.16 Process Update Event

Table 45: Process Description: Update Event

Identifier	UC-20: Update Event
Purpose	Allow admin to change details of an event
Requirements	WC_3770
Development Risks	None
Pre-conditions	Admin is logged in the system.
Post-conditions	Event's details are changed.

Table 46: Typical Course of Actions - Update Event: Success

Seq#	Actor's Action	System's Response
1	Admin asks system to update an event.	
2		System retrieves the event and displays a form for changing it.
3	Admin enters event's information and click on "update" button.	
4		System updates the event.

Table 47: Alternate Course of Actions - Update Event: Error

Seq#	Actor's Action	System's Response
1	Refer to typical course of action step 1-3	
2		System displays errors to admin in order to change the input.

2.1.3.1.17 Process Search Pictures

Table 48: Process Description: Search Pictures

Identifier	UC-21: Search Pictures
Purpose	Allow admin to search pictures.
Requirements	WC_3767
Development Risks	None
Pre-conditions	Admin is logged in the system.
Post-conditions	List of pictures is displayed to admin.

Table 49: Typical Course of Actions - Search Pictures

Seq#	Actor's Action	System's Response
1	Admin asks system to display a list of pictures.	
2		System retrieves pictures and display a list of them to admin. Also generates a form to filter the list based on that.
3	Admin enters form information to filter the list.	
4		System updates the list.

2.1.3.1.18 Process Restore Reported Picture

Table 50: Process Description: Restore Reported Picture

Identifier	UC-22: Restore Reported Picture
Purpose	Allow admin to restore a reported picture.
Requirements	WC_3774
Development Risks	None
Pre-conditions	Admin is logged in the system.
Post-conditions	A reported picture restores.

Table 51: Typical Course of Action - Restore Reported Picture

Seq#	Actor's Action	System's Response
1	Admin asks system to restore a reported picture.	
2		System retrieves pictures and restores is to the system.

2.1.3.1.19 Process Change App Settings

Table 52: Process Description: Change App Settings

Identifier	UC-23: Change App Settings
Purpose	Allow admin to configure system.
Requirements	WC_3772
Development Risks	None
Pre-conditions	Admin is logged in the system.
Post-conditions	System settings changes.

Table 53: Typical Course of Action - Change App Settings

Seq#	Actor's Action	System's Response
1	Admin asks system to show the system settings.	
2		System retrieves settings and displays them to admin in a form that admin can change.
3	Admin changes the settings.	
4		System stores the new settings.

2.1.3.1.20 Process Generate Statistics Report

Table 54: Process Description: Generate Statistics Report

Identifier	UC-24: Generate Statistics Report
Purpose	Admin generates some report about the system.
Requirements	WC_3773
Development Risks	None
Pre-conditions	Admin is logged in the system.
Post-conditions	System settings changes.

Table 55: Typical Course of Action - Generate Statistics Report

Seq#	Actor's Action	System's Response
1	Admin asks system	
2		System retrieves settings and displays them to admin in a form that admin can change.
3	Admin changes the settings.	
4		System stores the new settings.

2.1.3.1.21 Process Login

Table 56: Process Description: Login

Identifier	UC-1: Login
Purpose	Allow a user to log in the system.

Requirements	WC_3583
Development Risks	None
Pre-conditions	User needs an email account or Facebook account.
Post-conditions	User logs in the system.

Table 57: Typical Course of Action – Login: With Username: Success

Seq#	Actor's Action	System's Response
1	User fills Username and password fields with registered information and clicks the log in button.	
2		System checks if the username and password match with the registered information in the database and allows user to log in.

Table 58: Typical Course of Action – Login: With Username: Fail

Seq#	Actor's Action	System's Response
1	User fills Username and password fields with registered information and clicks the log in button.	
2		System checks if the username and password match with the registered information in the database and returns a proper message.

Table 59: Alternate Course of Actions - Login: With Facebook

Seq#	Actor's Action	System's Response
1	User clicks the log in with Facebook button.	
2		System directs user to Facebook app and asks him to log in.
3	User logs into Facebook and is automatically directed to the PicShare app's main screen.	
		System waits for an action from the user.

2.1.3.1.22 Process Register

Table 60: Process Description: Register

Identifier	UC-3: Register
Purpose	Allow a user to register the system.
Requirements	WC_3583
Development Risks	None
Pre-conditions	User needs an email account.
Post-conditions	User registers the system.

Table 61: Typical Course of Action – Register: With Email: Success

Seq#	Actor's Action	System's Response
1	User fills username, email, password, and confirm password fields and clicks the register button.	
2		System checks the information in the database and stores the information.

Table 62: Typical Course of Action – Register: With Email: Fail

Seq#	Actor's Action	System's Response
1	User fills username, email address, password and confirm password fields.	
2		System checks the information in the database and returns a proper message when it finds.

2.1.3.1.23 Process Logout

Table 63: Process Description: Register

Identifier	UC-2: Logout
Purpose	Allow a user to register the system.
Requirements	WC_3583
Development Risks	None
Pre-conditions	User is logged in.
Post-conditions	User gets logged out.

Table 64: Typical Course of Action – Logout

Seq#	Actor's Action	System's Response
1	User clicks on the logout button.	
2		System gets user logged out of the system and shows a success message.

2.1.3.1.24 Process View Nearby Pictures

Table 65: Process Description: View Nearby Pictures

Identifier	UC-16: Register
Purpose	Allow a user to view the pictures nearby.
Requirements	WC_3761
Development Risks	None
Pre-conditions	User logs in the system.
Post-conditions	A list of pictures is shown to user.

Table 66: Typical Course of Action – View Nearby Pictures

Seq#	Actor's Action	System's Response
1	User chooses the radius for browsing pictures in nearby locations.	
2		System retrieves the pictures in a given radius.

2.1.4 Modes of Operation

The PicShare System will not have multiple modes and operate in only one mode, so that no further description is required.

2.2 System Analysis Rationale

We have only one type of operational stakeholder: basically smart phone users between the age of 18-45, but it is not restricted with age. Users will need a smart phone and they must download the application. They also need to create an account to use the application. There is an an option to create an account with current Facebook account, or they can create it with an email address and a password.

The PicShare system provides users with the ability to share their picture to a nearby location or in a public event by writing a hashtag. The system allows users to create a private event with a password. So, only invited people can see the event and access it via password. In this way, multiple users at an event will be able to see the pictures taken from others, and they will be able to have the opportunity to download and store the pictures on their own devices.

3. Technology-dependent Model

3.1 Design Overview

3.1.1 System Structure

Figure 6: Software Component Class Diagram

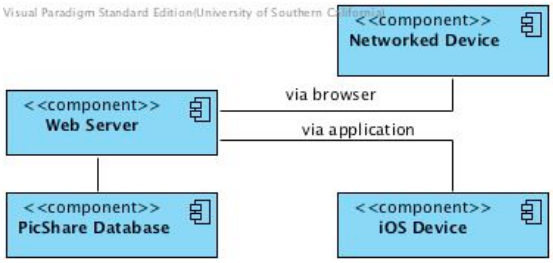
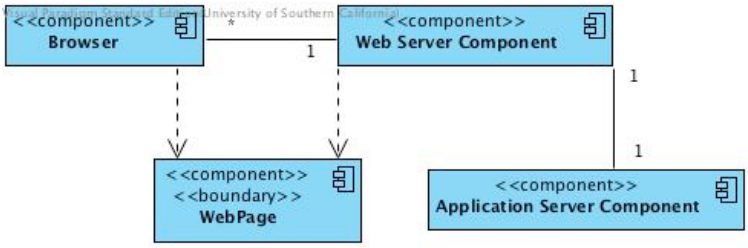


Figure 4: Hardware Component Class Diagram



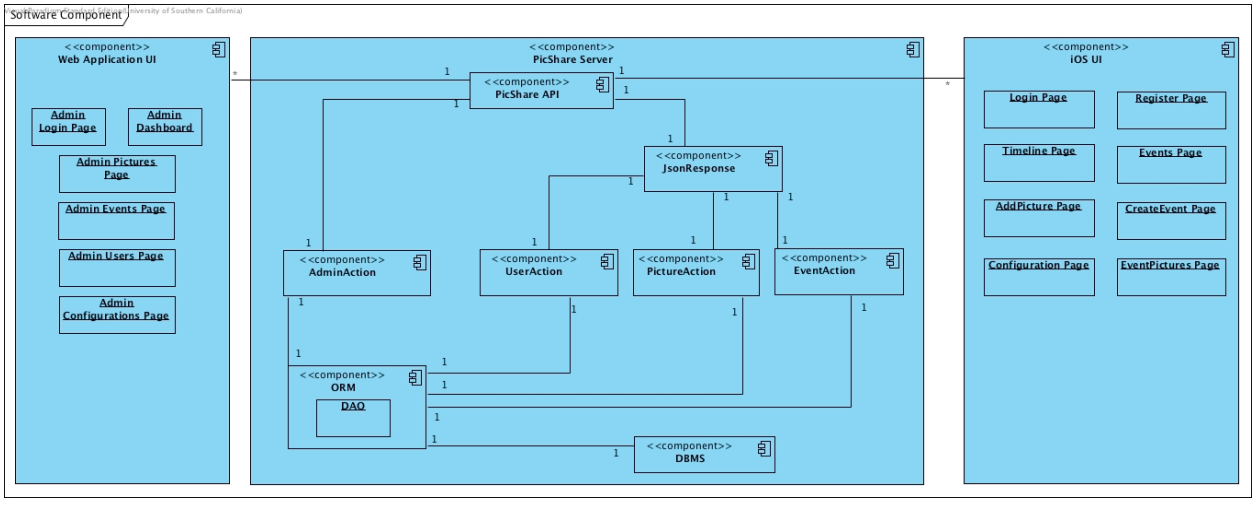


Table 67: Software Component Description

Hardware Component	Description
Networked Device	Any device that is connected to internet. Admin can open browser with that device and connect to admin module.
iOS Device	Devices that users use to connect to system.
Web Server	The server that our server side application will be running on it.
PicShare Database	Database of our system.

Table 68: Software Component Description

Software Component	Description
PicShare API	This is the component that UI components have interaction with. It provides a way of connecting to inner components of server.
Profile Management	A component that is used for profile and user management.
Event Management	A component that is used for event management.
Picture Management	A component that is used for picture management.
Data Access Layer	A component that is used for accessing the database. Implementation of all DAOs will be here.
DAO	Data Access Objects classes that are used for connecting to database.
Database	Represents the database of our system.
Login Page	Page for user login in iOS application.
Register Page	Page for user register in iOS application.
Timeline Page	Page for user timeline in iOS application. It contains a list of nearby pictures.
Add Picture Page	Page for adding a picture in iOS application.
Events Page	Page for events list in iOS application.
Configuration Page	Page for changing configurations in iOS application.
Create Event Page	Page for creating event in iOS application.
Event Pictures Page	Page for showing pictures of an event in iOS application.
Admin Login Page	Page for admin login in admin website.
Admin Dashboard	Page for showing an overall statistics in admin website.
Admin Pictures Page	Page for managing pictures in admin website.
Admin Events Page	Page for managing events in admin website.
Admin Users Page	Page for managing users in admin website.
Admin Configurations Page	Page for changing configurations in admin website.

Table 69: Web Framework Component Description

Web Framework Component	Description
Browser	An Internet browser that connects to the Volunteer Tracking System web application and is responsible for displaying Volunteer Tracking System web pages.
Web Server Component	The server component that routes all network traffic and requests between external systems and the application server.
Application Server Component	The server component where the Volunteer Tracking System resides on. All the logical computations are done on this component.
Web Pages	The actual web pages created by the Volunteer Tracking System.

3.1.2 Design Classes

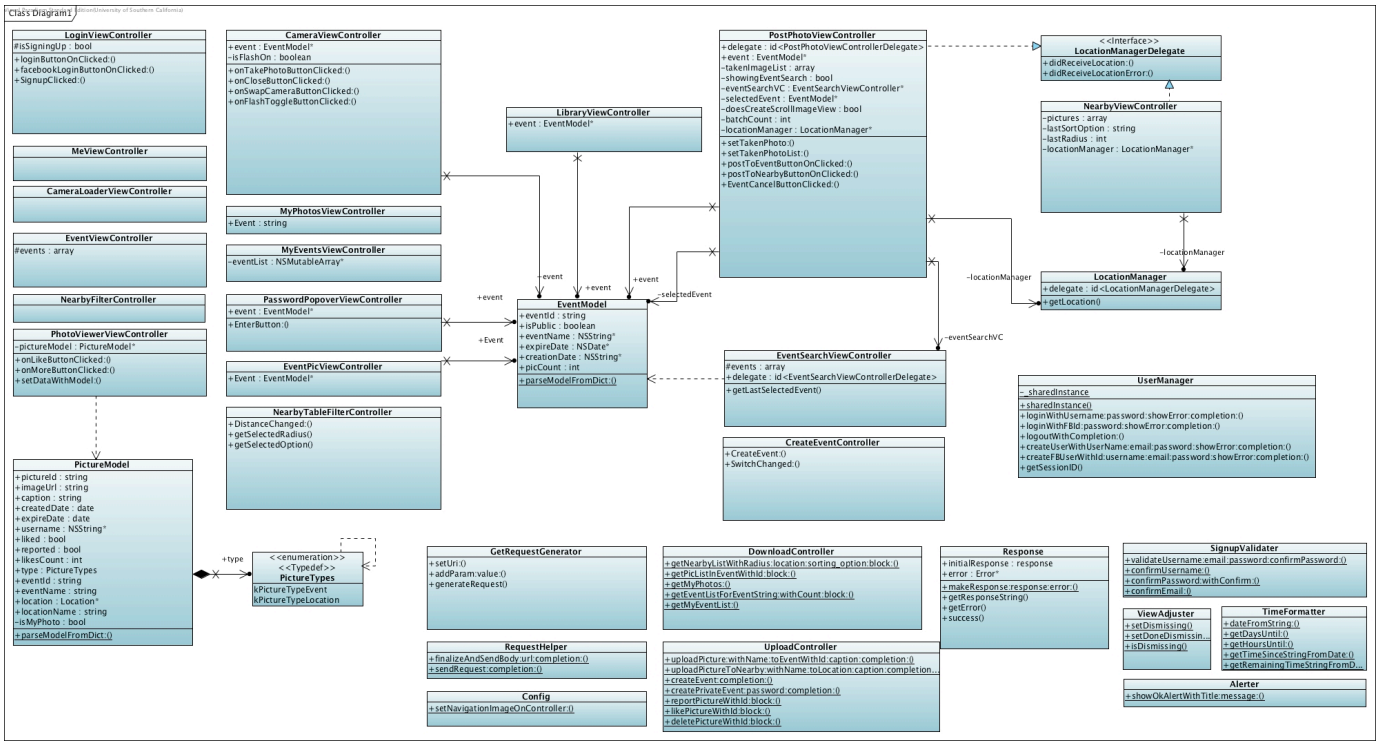


Figure 5: Design Class Diagram

3.1.3 Process Realization

In this section we created some sequence diagrams, to demonstrate some features that were not completely clear.

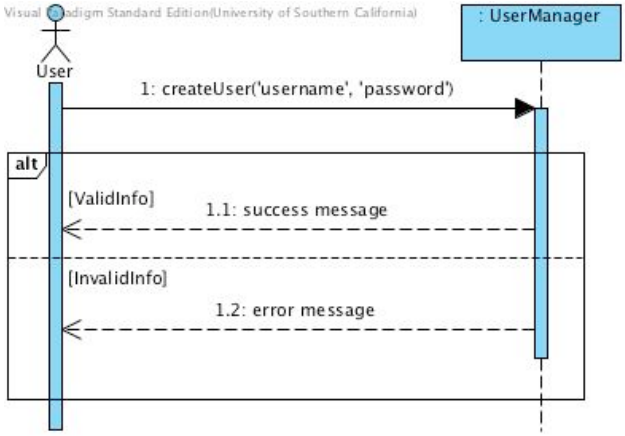


Figure 6: Sequence Diagram - Register with Email

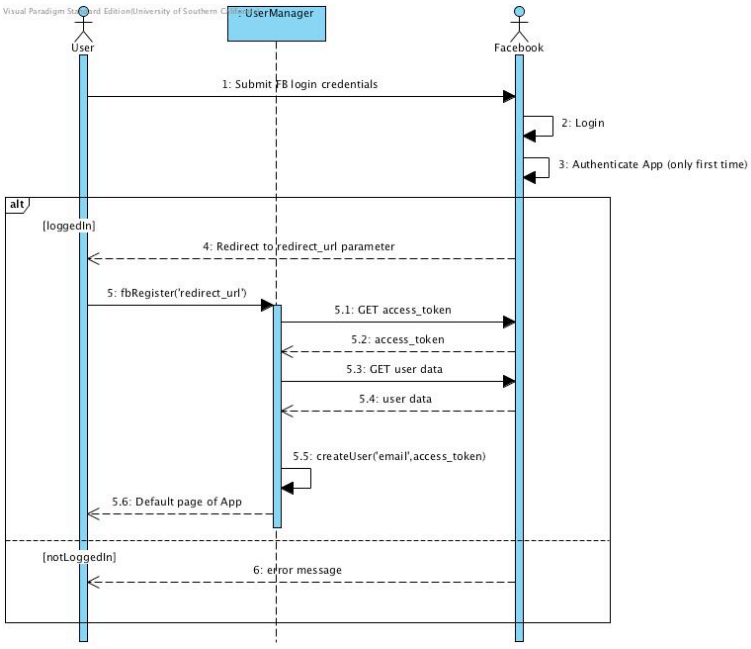


Figure 7: Sequence Diagram - Register with Facebook

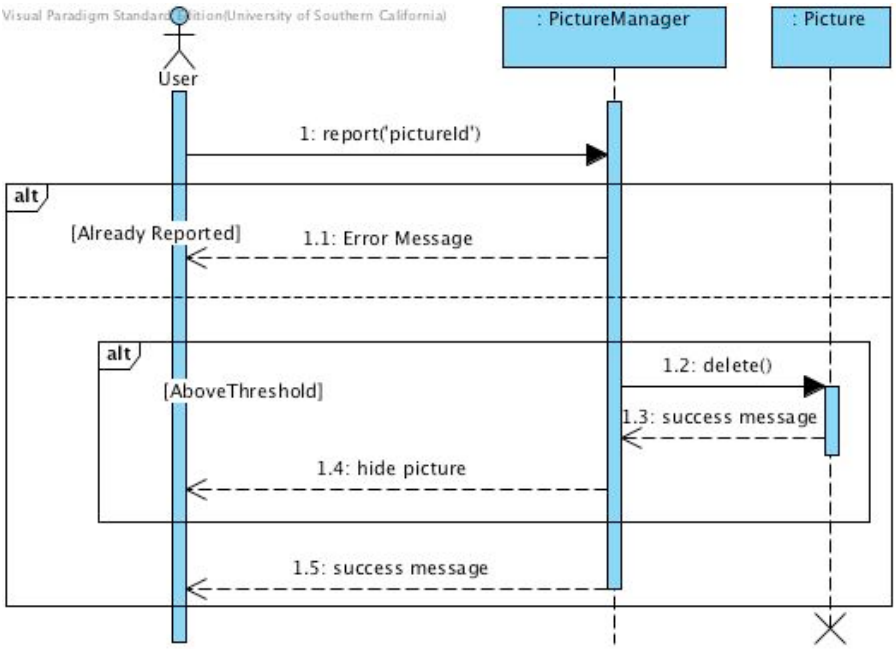


Figure 8: Sequence Diagram - Report Picture

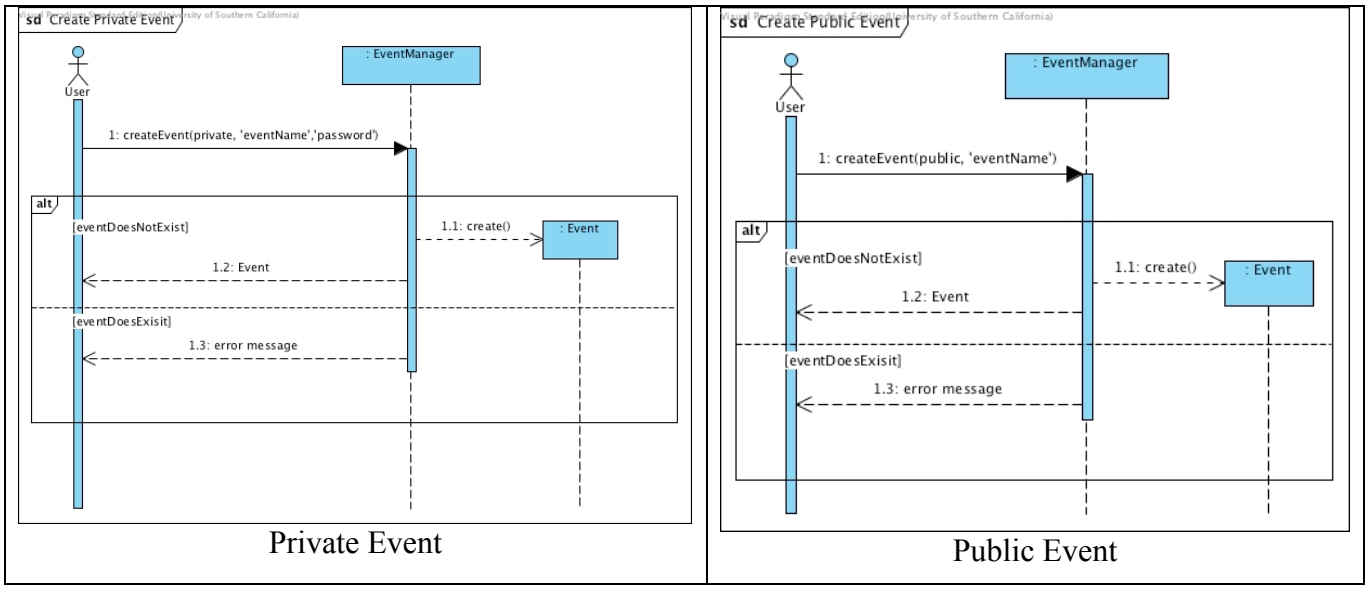


Figure 9: Sequence Diagram - Create Public and Private Event

3.1.4 Entity Relationship Diagram

We wanted to make sure that we have a complete understanding of our system, so we decided to create entity relationship diagram.

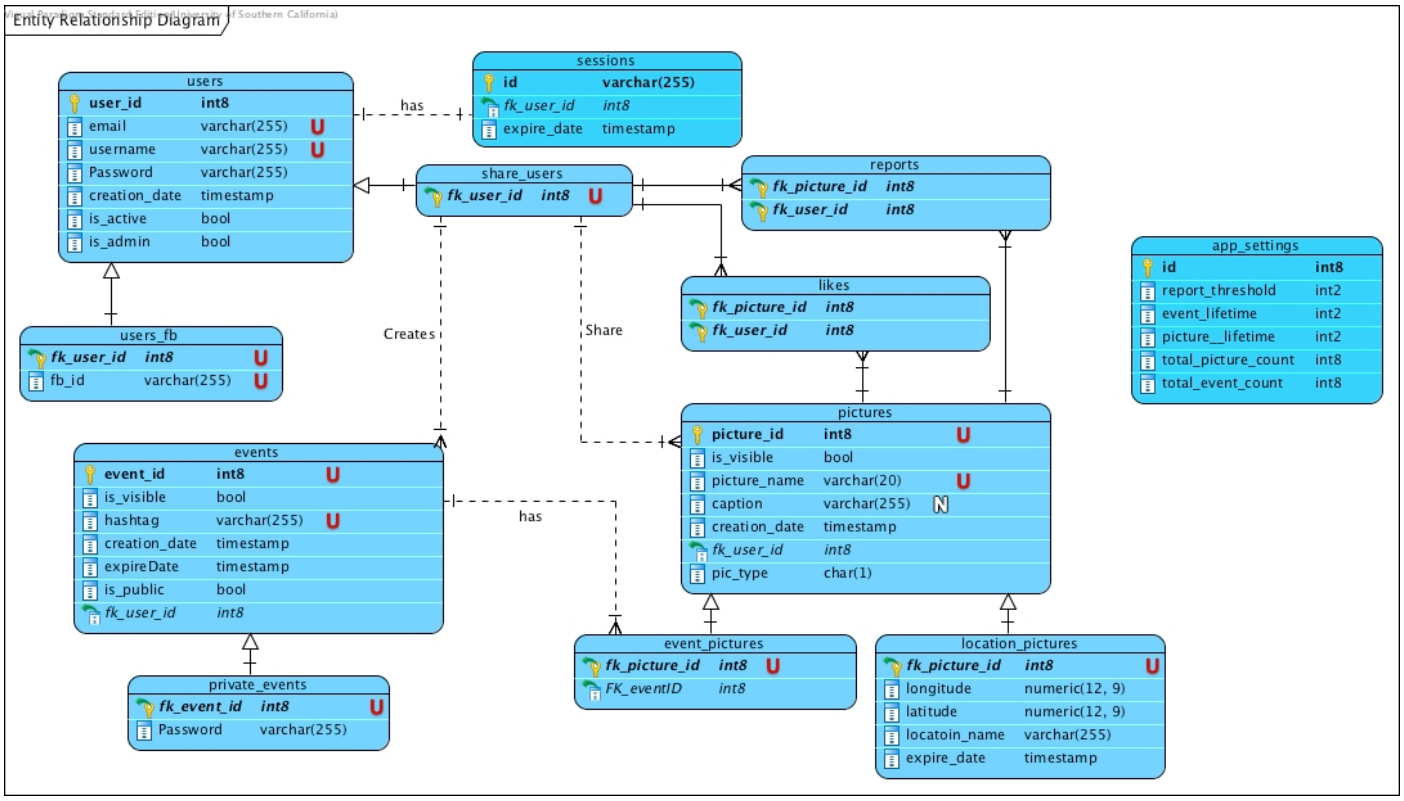


Figure 10: Entity Relationship Diagram

3.2 Design Rationale

We adapted three-tier architecture because our app will have basically presentation tier as user interface, application tier as business logic and all functionalities, and data tier as database processes.

- Presentation Tier
  - User Interface component
- Application Tier
  - Profile Management Component
  - Picture Management Component
  - Event Management Component
- Database Management System

The User Interface Component defines all parts of the interface of the application. The user will see main pages and the styles of the application. iOS SDK will provide the foundation tools for this.

The Application Tier includes the class, use case and sequence diagrams that compose the logic of all functionalities for the system. All the business logic will be processed at the background. We chose these diagrams because it's easy to show the relation between processes. With profile management component, the system will be able to create an account for the user by using username and password. Also login and logout functions will be provided to user. Picture management component will include all processes about pictures such as posting a picture, reporting a picture, deleting a picture, saving a picture to device etc. Creating private/public events, deleting private events, saving hashtags are the functionalities for the event management component.

The data tier is the place for storing and retrieving information from a database. It provides access to the data. Retrieved information from the database is passed back to the application tier for processing, then eventually to the end user. We chose to use MySQL because it's widely used and easy to find documentation on the web. It's also compatible with HostGator which we will use for keeping our database files and server.

4. Architectural Styles, Patterns and Frameworks

In the table below, you can see list of architectural styles, patterns and frameworks that we used so far.

Table 70: Architectural Styles, Patterns, and Frameworks

Name	Description	Benefits, Costs, and Limitations
Three-tier architecture	We have presentation tier, application tier and database management system layer.	With this, we can be able to make changes to each of these layers, without the need to the change other layers; as long as the interface between them does not change.
MVC	Having models, views and controls separated.	It makes understanding of how code works easier and therefore more maintainable. Also, since our models are separate than other parts we can change them easier and without the need to change other parts.
Slim framework	It is a micro framework, that using this, we can create a PHP server easier.	It helps with the MVC pattern, since using this our model, view, and controls are in different places.