**** 

**Bilkent University**

**CS 491**

**Senior Design Project I**

**Fall 2013**

**On Fire: No Lie, See Live**

***High-Level Design Report***

December 31th, 2013

Group Members
Baha Uluğ | Barış Şahin | Bengisu Şahin | Berkin Şansal | Sıla Şirin

Supervisor
Fazlı Can

Jury Members
Tuğrul Dayar | Hakan Ferhatosmanoğlu

Innovation Expert
 Serkan Köse

Contents

[List of Figures iv](#_Toc371978419)

[1. Introduction 5](#_Toc371978420)

[2. Current Systems 5](#_Toc371978421)

[3. Proposed System 7](#_Toc371978422)

[3.1. Overview 7](#_Toc371978423)

[3.2. Functional Requirements 8](#_Toc371978424)

[3.3. Non-Functional Requirements 9](#_Toc371978425)

[3.4. Pseudo Requirements 10](#_Toc371978426)

[3.5. System Models 10](#_Toc371978427)

[3.5.1. Scenarios 10](#_Toc371978428)

[Scenario 1: Create a profile 10](#_Toc371978429)

[Scenario 2: Fire up 10](#_Toc371978430)

[Scenario 3: Add friend 11](#_Toc371978431)

[Scenario 4: Become an owner 11](#_Toc371978432)

[3.5.2. Use Case Model 12](#_Toc371978433)

[3.5.3. Object and Class Model 13](#_Toc371978434)

[3.5.4. Dynamic Models 15](#_Toc371978435)

[3.5.4.1. State Diagrams 15](#_Toc371978436)

[3.5.4.2. Activity Diagrams 16](#_Toc371978437)

[Fire up and add video 16](#_Toc371978438)

[Search and add a friend 16](#_Toc371978439)

[3.5.4.3. Sequence Diagrams 17](#_Toc371978440)

[Scenario 1 17](#_Toc371978441)

[Scenario 2 17](#_Toc371978442)

[Scenario 3 18](#_Toc371978443)

[Scenario 4 18](#_Toc371978444)

[3.5.5. User Interface – Navigational Paths and Screen Mock-Ups 19](#_Toc371978445)

[Scenario 1 19](#_Toc371978446)

[Scenario 2 20](#_Toc371978447)

[Scenario 3 21](#_Toc371978448)

[4. Conclusion 22](#_Toc371978449)

[5. Glossary 22](#_Toc371978450)

[6. Schedule 23](#_Toc371978451)

[7. References 23](#_Toc371978452)

# List of Figures

[Figure 1 : Foursquare screenshots. 6](#_Toc371976453)

[Figure 2 : AroundMe screenshots. 7](#_Toc371976454)

[Figure 3 : Use case diagram. 12](#_Toc371976455)

[Figure 4 : MVC model. 13](#_Toc371976456)

[Figure 5 : Class diagram. 14](#_Toc371976457)

[Figure 6 : State diagram for possible screens. 15](#_Toc371976458)

[Figure 7 : Activity diagram for adding a video. 16](#_Toc371976459)

[Figure 8 : Activity diagram for searching and adding a friend. 16](#_Toc371976460)

[Figure 9 : Sequence diagram for scenario 1. 17](#_Toc371976461)

[Figure 10 : Sequence diagram for scenario 2. 17](#_Toc371976462)

[Figure 11 : Sequence diagram for scenario 3. 18](#_Toc371976463)

[Figure 12 : Sequence diagram for scenario 4. 18](#_Toc371976464)

# Introduction

By the enhancement of Smart Phones and Internet speed, users can get any information when they need it. However, because of the information pollution, it is very hard to get exact and true information about the desired search. Especially, in social media there is too much misleading information that leads misconceptions [9]. Our project aims to solve this problem on location-based social media.

As a solution of this problem, our senior project will be a mobile application that allows users to see what is happening in a place in a particular moment by video sharing with social media integration from users and owners of the places. Therefore, videos are the best way to give some idea about a place; users could not mislead by other users.

The link to our project website: <http://supportonfire.wix.com/onfire>

## Purpose of the System

Our application will allow users to have an idea about the places according to shared videos. For owner of the place, application will allow adding an advertisement video to introduce the place on the place’s page. This feature will be very beneficial for owners because they will be able to advertise their place to their target population. On the other hand, users can see and search place’s and users’ locations on the map, the camera, or a basic list and also name of the places and users can be seen in any option. Users can fire up and see their connections’ fires in order to show which places they are hanging out.

## Design Goals

## Definitions, Acronyms, and Abbreviations

## Overview

# Current Software Architectures

* **Foursquare [5]:** Foursquare makes people check in any place. With this app, they can also reach their friends check-in. By this way, they can go a place which got many likes or has a good recommendations.

Our program and Foursquare have mainly similar concept in terms of check-in. In addition to Foursquare, our program is unique on uploading videos at a particular moment. By this way, videos will have a role which will lead a place to come into prominence.



Figure 1 : Foursquare screenshots.

* **Around Me [6]:** This application shows you nearby places and also places according to search results. There are categories to filter the results. You can see nearby places in a list, in a map, or in a camera.

This program shows places on camera by augmented reality. We will have a feature that which will show places and users. By launching the camera in the app, nearby places and people will be appear according to their locations. Location-based augmented reality technology will help us to improve this [4]. Your friends, other crowd, and favorite places will be under your hand with their accurate locations and directions.



Figure 2 : AroundMe screenshots.

# Proposed Software Architecture

## Overview

Onfire will be a mobile application for people who like to explore new places by sharing and watching videos about a place at a particular moment. Also, with social media integration and augmented reality, system will help users to share their locations and see their distances from their friends.

 Some technical details about Onfire are listed below.

* **Platform:** iOS [7]
* **Language:** Objective-C [7]
* **Architecture:** MVC
* **Design Patterns:** Singleton
* **Database:** Amazon S3 [3]

## Subsystem Decomposition



Figure 3 : Context Diagram

Subsystems of MVC architecture are shown below:



Figure 4 : Subsytem Decomposition Diagram

## Hardware / Software Mapping

## Persistent Data Management

## Access Control and Security

## **Global Software Control**

OnFire is an event driven mobile application. It requires authentication by using social network account or by just signing up the OnFire. The program needs the Internet connection to get updates and to be used properly.

 When user clicks OnFire’s icon and runs the application, Application Management waits for the e-mail address and password from the user. After, user successfully login to the system, he/she can start to use the OnFire.

 If user wants to add a video, then Event Management is invoked, and it waits for the user to complete the process. When user clicks *“Fire Up”* button on the screen, the system starts to process the video. For this process, the video, the particular place and the event related to it are the inputs. Additionally, Security and Transaction Management is invoked as well, and the video is attached to the related event. In this way, the database is updated accordingly.

 Finally, when the user desires to update his/her settings, then Application Management calls the User Management. During this process, user’s account is updated according to the setting preferences of the user.

## Boundary Conditions

**Initialization:** Initialization condition for OnFire is a smart phone or tablet which has IOS operating system. When user runs the application, the system initializes itself if the device is connected to a wireless network. Then, login page is presented to the user at startup time. So as to use OnFire, user needs to sign up the system or have Facebook account. Login page will not appear again until user logouts on the current device.

**Termination:** When OnFire is opened, if an attempt occurs to going home page of device, application will run in the background with restricted functions. If application removes from the device, it will terminate itself. Additionally, if the device loses wireless connection or the database crashes, then the system terminates itself as well. Nonetheless, if the user does not have IOS operating system, he/she cannot use OnFire.

**Failure:** In terms of software failures, one of the failures is related to the version of the IOS operating system; if user has IOS versions before IOS 4, he/she cannot use the application and fail. Other problem can occur because of user’s Facebook account. These kinds of problems are not related to our application.

In terms of hardware failures, if the camera of the device does not work properly, the system gives an error message to the user. In this way, user cannot use the features related to camera. However, he/she can use the other features of OnFire. On the other hand, if device battery is quite low to run application or database crashes, the application terminates itself.

# System Services

* **Model:** Model layer is where application logic, connection with database and network are achieved.
* **Business:** This package is responsible for business logic of the system. They provide services for controller classes.
* **Application Management:** Includes functions for general application logic like sign up, login and settings.
* **User Management:** Includes functions that will be needed after user logged into the system.
* **Event Management:** Includes functions that are specific to events.
* **IO Management**: Includes IO functions.
* **Database:** This package is where database management and recovery management are achieved.
* **Database Management**: Responsible for database connections and software representations of database objects.
* **Recovery Management**: Responsible for back-up systems.
* **Network:** This package is where network management, transaction management and security management is achieved.
* **Network Management:** Responsible for network connection.
* **Transaction Management:** Responsible for all communication between system and database.
* **Security Management:** Responsible for transaction security.
* **View**: All user interface elements reside here.
* **UI Management**: Responsible for user interface elements.
* **Controller**: All controller classes reside here for each view.

# **Conclusion**

 As a result, our project is a social network application, which enables users to see what is happening in a place in a particular moment by video sharing with social media integration from users and owners of the places. In this report, we prepared clear class diagram and four scenarios were selected in order to explain the flow of application in the best way. In this way, we mentioned about current systems similar to OnFire’s functional and nonfunctional requirements, use case, class, state, activity and sequence diagrams with four scenarios, mockups related to these scenarios.

# **Glossary**

|  |  |
| --- | --- |
| **Keyword** | **Definition** |
| Fire | Name for the activity which made in a place |
| Fire up | The verb form of fire |
| Video | Video is used in each fire |
| Description | Explanation either for places or for fires |
| Comment | Users’ idea either for places or for fires |
| Like | Positive feedback either for places or for fires |
| Dislike | Negative feedback for places |
| Nearby Places | Places around a specific location |
| Favorite | Adding a place to users special list |
| Member | Who can fire up and use app with its main purposes |
| Owner | Who can add a place to his place list |
| Admin | Who controls and manages content by users and owners |

# Schedule

* Project Specifications - Monday, Oct. 7 2013, **Done**
* Analysis Report - Monday, Nov. 4, 2013, **Done**
* High-Level Design Report - Friday, Dec. 31, 2013, **This Report**
* Low-Level Design Report - Monday, Feb. 17, 2014, **Will be done**
* Final Report - Thursday, Apr 30, 2014, **Will be done**
* Presentations & Demonstrations - May 5 - 9, 2014, **Will be done**

# References

[1] . (2011). *Apple iPhone 4S* [Online]. Available:
<http://www.geekandhype.com/apple-iphone-4s-8830/iphone-4s-white/>

[2] . (2013). *Foursquare API* [Online]. Available:
<https://developer.foursquare.com/>

[3] . (2013). *Web, Mobile, and Social Apps on AWS* [Online]. Available: <http://aws.amazon.com/web-mobile-social/>

[4] . (2013). *Augmented Reality iOS Tutorial: Location Based* [Online]. Available: <http://www.raywenderlich.com/42266/augmented-reality-ios-tutorial-location-based>

[5] . (2013). *Foursquare* [Online]. Available:
<https://itunes.apple.com/tr/app/foursquare-find-restaurants/id306934924?mt=8>

[6] . (2013). *Around Me* [Online]. Available: <https://itunes.apple.com/us/app/aroundme/id290051590?mt=8>

[7] . (2013). *Xcode* [Online]. Available:
<https://developer.apple.com/xcode/>

[8] . (2013). *Facebook SDK for iOS* [Online]. Available: <https://developers.facebook.com/docs/ios/>

[9] M. J. Dollinger. (2013). *Business Horizons* [Journal]