

Waterfront Film Festival Project



Design Document for iPhone Application



Team Members:

Jordan Carney
Corey Bennett
James Watthanasintham

February 4, 2013

Overview

The team is tasked with producing an iPhone app for the Waterfront Film Festival (WFF). This design document will describe the methodologies and tools that will be used to flesh out ideas and develop the software application.

Implementation

Tools and Development

The Waterfront Film Festival iPhone App will be programmed using Objective-C and the Xcode development environment -- this is the proprietary language and software that needs to be used in order to develop for Apple products. Apple defines standards that every Apple software developer must conform to in documents on their developer website.

The WFF app will make use of various APIs such as Apple Maps and the Core Data database management provided by Apple. Additional APIs that will be used are Flickr for photos covering the festival and possibly Passbook for ticket implementation. Many iOS standard frameworks will be included in the project. This includes, but is not limited to, the Twitter, Facebook, and Message frameworks.

Code Repository

The code repository will be organized utilizing GitHub. A central repository for the project has been created. The team members will be able to pull data from the repo and make modifications. They will then be able to push the changes (via a "Pull Request") to the central repo for approval from the repo administrator, Jordan. This is a

generic software development setup with new software developers to language. It prevents a surplus of merges and issues for other repository users. It also allows Jordan the ability to glance of team member code before submission and make sure it is good code, or the best code possible if he is feeling hyper-critical.

Testing

A majority of the testing will involve using the built-in iPhone simulator in Xcode and could be considered alpha testing. Beta testing will take place on an actual iPhone device -- a personal device or one that has been furnished for the team. We will test mainly on iOS 6, but we will also test on iOS 4 and 5 to ensure backwards compatibility. This will be done mainly on the iPhone simulator, but the group will have access to older iDevices with less updated iterations of iOS.

We will also conduct a series of usability tests on randomly selected people. These people will fit into the category that typically have mobile devices and would attend the Waterfront Film Festivals. We will get the analytics of attendees from past Waterfront Film Festival analytics if possible.

Division of Labor

Jordan will be responsible for upholding software integrity. He will oversee the application development process and ensure that steps are taken for efficient project completion and performance. Jordan will also work on the base setup of the application and big chunk of the major, more complex features such as map integration, dealing with assets, and the venues portion of the application. He will also delegate specific portions of the application to his teammates who are learning Objective-C for the first time.

James will be working on the “about” pages and the contact page. These will be generic information pages that will need to be cohesively designed. That is why one person will be assigned to them. James will also be working on setting up the API with the rest of the team. This will be a strongly collaborated element of the project.

Corey will be working on the ticketing system and subsequent pages involved in that process. Corey will also be working on a small portion of the mapping system. This will involve providing the user walking/driving directions to the event from their current location. The map will also show the locations of the different venues and Corey will be involved in implementing that as well.

Database

We’ll be using Apple’s Core Data to manage the databases our app uses to store data for information regarding films, venues, lodging, and images. A lot of this data will need to be pulled from a server to maintain the dynamic nature of the material. It needs to be dynamic because event information changes every year and new (late) additions also need to be accounted for in the application. See figure 1.1 - 1.3 below for a ER representation of the proposed database.

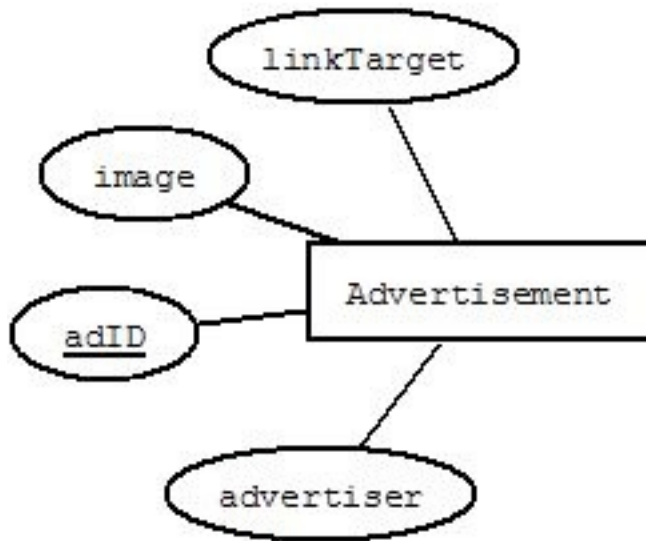


Figure 1.1: Advertisement entity of the database

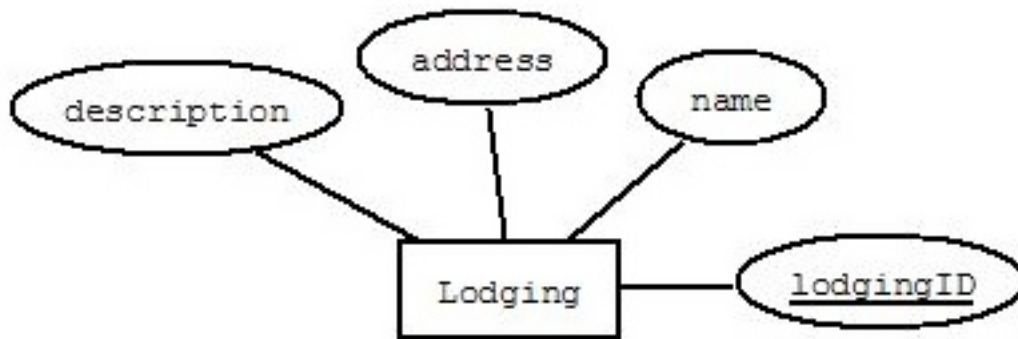


Figure 1.2: Lodging entity of the database

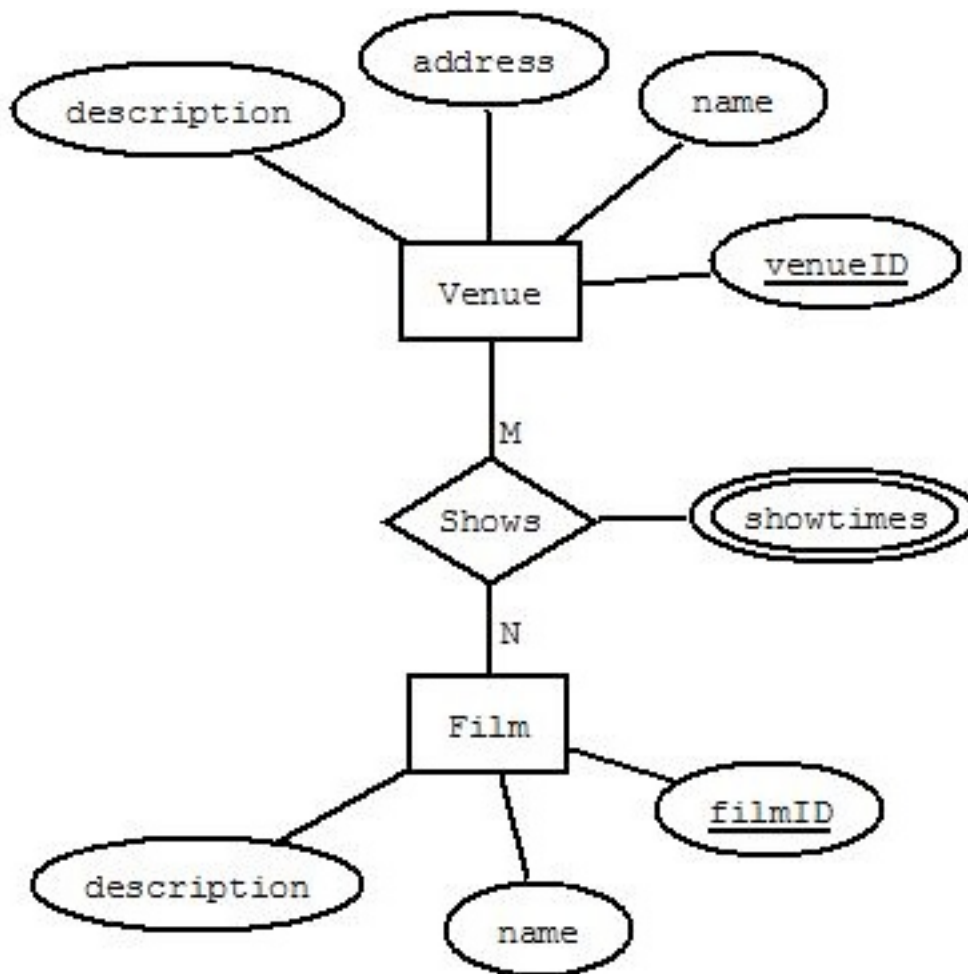


Figure 1.3: Venue and film entities of the database

Client/Server

We will be setting up an API for the Waterfront Film Festival. They do not have any web servers set up to pull information from; thus, we will be doing that as well. This server will handle requests for information sent from the iPhone application. This will allow event information to be dynamically updated and used for many years to come.

We will be using the Django server side web framework to store the data and send data when requested. Jordan is familiar with this framework and has not had any issues. That is the reason for this particular implementation.

User interface

Balsamiq Prototypes:



Go to page 8



The user interface of this application is kept relatively simple. Notice, throughout the prototypes, the tabbed bar icons change (the bottom bar with icons); this is a design aspect we changed. Changing tabbed bar icons did not offer a intuitive navigation experience. The user would never be sure what is actually on the tabbed bar.

Notice the main screen upon loading the application has the most important elements as buttons on the view. This will give the user immediate access to the desired functionality of the application. Users are more likely to press what is clearly visible in front of them than what is not, obviously.

The advertisement is placed front and center of the main page as well because the Waterfront Film Festival will be making money from this application through advertisement of their sponsors. The more visible or touchable the ads are, the more likely people will be enticed by them. This was essentially a business decision.

UITableViews are used for a lot of different things such as the different venues and the movies playing at them. This is consistent throughout and the user will come to expect information presented in an easily navigable, scrollable, and interactive table.

Application Rationale

We want a relatively simple, streamlined app. Something that fulfills the client's desires while optimizing usability and performance.

This will essentially be a marketing application. The purpose is to get people involved in the event and know more about what is going on. Ideally, it will allow an easy median for purchasing tickets and planning attendance to the event. It will also navigate attendees to where they need to go; this will ensure a seamless experience. It will also generate ad revenue for the Waterfront Film Festival.

The functionality of this application is not overly complex. It is a series of small functions that will need to be done well. We initially were

presented prototypes with a plethora of buttons and confusion. We will aim to minimize the clutter and accentuate the few functions that are necessary and most important.

We initially presented the idea of doing an Android application as well, but due to time constraints we had to cancel this plan. This will allow the iOS application to be more profound, fluid, and more sufficiently developed.

Making this application support Passbook was also considered. This functionality is not going to work with the current state of the ticketing system and will not be implemented fully, if at all.

Design Methodology

We are going to be utilizing the Agile Development process. This will include 2-week sprints where specific tasks are to be completed. Each member of the team will have their own responsibility during these two weeks and each sprint will always end with a meeting to collaborate the work that was done. If the work was not completed, that will be pushed to the next sprint where it will be completed in conjunction with that sprint's agenda.

See the next page for our project timeline utilizing a Gantt chart.

