AirCaddy – Final Report

Created by grant folgate, Eric byrd, madison wiecorzek, and ryan vero

2017

MODULE 1 – ORGANIZATION DESCRIPTION

* 1. Organization Description
     1. Mission, Objectives, Industry
     2. Information Systems Culture of the Organization
  2. Overview Decription of the Organization Business Unit
  3. Business Area Analysis of the Unit

MODULE 2 – PROJECT DEFINITION

* 1. **General Application Goal**

Original Proposal

There are various golf based application areas that are not established yet and for which new ideas and models are presently under development. For this project, the primary user is assumed to have some interest in the sport of golf and the strategic process of the game itself. Many avid golfers currently experience difficulties when navigating a new course. They have to rely on a map at each tee box with minimal information on each hole. A golfer using this application can gain a valuable perspective in terms of blind spots, depth perception, and a feel for the terrain they are about to encounter.

Project Goal

Provide an application that gives the user a preview of the course hole with a drone view to improve overall golf experience and allow for a social media platform for users to share feedback on course/hole experiences.

Environment and Boundaries

The application environment will be primarily at the golf course of choice by the user, although the application would work anywhere that an internet connection is available, either through cell data or through Wi-Fi.

* 1. **Statement of Work**

***2.2.01 Objectives of the Project***

Our objective for this project is after a year of being on the market, we plan to have a growing user rate of 20% quarterly and increase the number of golf courses available in the application by 55% by the end of the year.

***2.2.02 Success Criteria***

How many users of the application and how often the users actually use the application will determine success. Usage will be tracked through footage view counters and commented reviews. The most important success factor for this project is solidifying a functionality that does not exist in any other golf application. The unique feature in our golf application is the use of state of the art drone footage that will be provided at each hole of the course that will be broadcasted in our application. Externally, the more courses and users that apply to add footage and comments on our application, the greater the change of market growth.

***2.2.03 Risk Analysis***

There will be some marketplace competition including the following existing applications 18Birdies, GolfNow, and Ping. These applications do not provide the in-person view or depth charts but they do offer an aerial view of each hole and the yardage to the hole; this could be enough information for some users. User acceptance can be another issue because some users will want to use the application exclusively while they are on the course; there is no guarantee that the golf course area will have adequate 4G coverage to access the application. The highest risk we have is staff illness, since we have a small team of developers, if any member became ill this could potentially setback the delivery date.

MODULE 3 – PROCESS MODELING

* 1. ***General Application Goal***

To help better describe this section, we will provide a user story diagram to give a better picture of how the users correspond to the functionalities offered by the system. With each of these functionalities a system sequence diagram will be provide to better explain the flow of how the system will respond.

***\*Upload image of the user story diagram\****

* 1. **Statement of Work**

In this section we will define all our listed functionality described in the user story diagram. In each functional requirement we will describe the inputs that go into them, the outputs, and the flow of data for each of them.

1. ***Request ability to be able to manage course footage***

The authorized type of user will fill out a form that will require them to provide the name of the course they are trying to upload for, their phone number, their email, their name, and justification on why they would like to upload course footage. By doing this we can reduce the risk of bad video data being uploaded to our application. This information will be available for an administrator to approve. The outputs of this functional requirement will either be a rejection and the user will be unable to upload course footage or they will be granted authorization to upload video footage of their course.

1. ***See golf course review data***

Any type of user will be able to view course review data that each course uploaded to the application may or may not have. To generate this data, once a course is uploaded to the application; our application will use the Yelp Fusion API to gather high level review data about the course which will then be displayed to the user to give better insight to them on whether it is worth exploring that course in more detail.

1. ***See high level course information***

Any type of user will be able to view high level course information that includes the operational hours of the golf course, address, and the phone number. To generate this data, once a course is uploaded to the application; our application will use the Yelp Fusion API to gather general business information which will then be displayed to the user which will allow the user to contact the course owner if they have more questions.

1. ***See footage per hole of golf course***

Unauthorized and authorized users will want to have this capability to give them better insight on what a particular hole is like on that golf course. To do this, the user will click on a golf course and select a hole they would like to see. To generate the video for the user to see, our application will use YouTube’s API to show the particular hole on the particular course the user would like to view.

1. ***See difficulty/comments per hole***

Any type of user can view this data and it will be generated by authenticated users to post what they think about that course by giving it a difficulty rating and providing any useful feedback that could help the user. This type of data will be obtained from the application’s database which is a Microsoft SQL Server instance.

1. ***Upload difficulty/comments per hole***

Authorized users will have privilege to do this. This type of user will fill out a form that requires them to give the hole a difficulty rating (1 being very easy, and 10 being very difficult) and some general comments about the hole. This information will be served in our application’s Microsoft SQL Server database instance.

1. ***Approve authorized users to upload course footage and become Golf Course Owner type user***

An administrator will be able to review all of these types of requests and have the ability to reject or accept that type of user to do those kinds of actions. This information will be obtained from our application’s Microsoft SQL Server database instance.

1. ***Manage course footage for their course***

A Golf Course Owner type of user will be able to upload footage per course simply by clicking on the golf course they own, selecting a hole, and then uploading a video file. The metadata of the file will be stored in our application’s Microsoft SQL Server database instance (File name, YouTube Id to obtain from YouTube), and the video itself will be stored on YouTube utilizing their API.

MODULE 4 – DATA MODELING

4.1 Initial ERD for the Application

4.2 Data Dictionary

MODULE 5 – DEVELOPMENT STRUCTURE

5.1 Work Breakdown Structure

|  |  |
| --- | --- |
| **Request ability to be able to manage course footage** | * Create web form for administrator approval * Provide UI to upload video |
| **See golf course review data** | * Create Yelp Fusion API Client * Create UI to show golf course review rating * On UI show user’s reviews |
| **See high level course information** | * Display course information from Yelp API |
| **See footage per hole of golf course**  **See difficulty/comments per hole** | * Display footage of selected hole from YouTube API * Display user comments from SQL database |
| **Upload difficulty/comments per hole** | * Display form to user, information collected will be stored in SQL database |
| **Approve authorized users to upload course footage and become Golf Course Owner type user** | * Retrieve user requests from SQL database, approve or deny |
| **Manage course footage for their course** | * Display form, ability to edit Database information and overwrite video file per hole of their own course. |

5.2 Planned Development Platform and Required Resources

For this application we plan on developing in the .NET Environment in which we will specifically be using ASP.NET to develop this web application. The backend which will be providing the logic to interact with our database and external web services will be created in the C# programming language. Our front end which will be used to display data and allow the user to interact with our application will be made using HTML, CSS, and JavaScript. To help speed up the CSS development we will be using the Bootstrap framework to provide a readily available set of styling to our application. To help provide a more robust user interface we will be using Knockout.JS which will help us provide 2-way databinding.

Our application will also require several more services to run correctly. A SQL Server database instance which will store our data necessary for the application to run. Our application will also interact with two external web services; one of which being the Yelp Fusion API which will provide our application with review data for each golf course entered into the system; the second web service we will be using is the YouTube API which will allow us to upload our videos to our private YouTube channel and allow us to play these videos within our application.

Lastly, this application will be deployed as a prototype on one of the servers held at Gannon that has at least Windows Server 2012 on it. On that server, we will need Internet Information Services enabled on it and it will need it to support version 4.5 of .NET or higher so our application can run correctly on the server. One of the servers that is a potential for our group to use is the pendragon server that is maintained by Professor Canell and Professor Coffman. Another potential server that can be available for us to use as a prototype is one of the servers maintained by ITS. Carol Kugler is a point of contact for this.