

# **Education Office Application Design and Development**

Jamie E. Johnson

Kennedy Space Center

Major: MS of Computer Science

Program: USRP Summer Session

Date: 29, July 2013

# Education Office Application Design and Development

J. Johnson<sup>1</sup>

*Florida &M University, Tallahassee, FL, 32307*

The content of this project focuses on designing and implementing a new prototype website for the Kennedy Intern Tracking System (KITS). The goal of the new website is to allow the user to search for interns based on several different categories and fields. In hence, making it easier to find a count of interns matching a set of criteria. The KSC office of education is the primary users of KITS, their job is to recruit interns year-round. As a secondary goal, each user will be able to generate a report of their searches onto a portable document format (PDF) file. The results of each search will be set to a limited amount per page. This site will be used for Kennedy Space Center internal purposes only. After the implementations are done, a visual walk through using screen shots will be used to guide the users through all of the different scenarios that are likely to occur when the users are navigating through the site. In addition, a demo of the site will be presented to the KSC Office of Education. JavaScript and JQuery are the languages that will focus on the functionality of the implementation. Hyper Text Markup Language will be used to form the foundation for the body structure of the website. Ruby will be the programming language used to elevate the prototype to a dynamic website and enable the programmer to finish with in an efficient time frame. Cascading Style Sheet will be the language used for the design and styling purposes. Rails is the framework that the new website will be built upon. By default, the database will be managed by Sequel Lite (SQLite). All users will need to be granted special privileges in order to use the site.

## I. Introduction

The KSC intern tracking System (KITS) is a system dedicated to collecting and managing information about National Aeronautics and Space Administration (NASA) KSC interns. KITS currently has a site available for the internet and intranet. The internet connects millions of computers and mobile devices through a global network in order for users to exchange information between each other. KITS uses the internet to host a website that allows its users (interns) to key in personal information to acquire a KSC visitor badge. Similar to the internet the intranet also uses a network that allows users with computers and mobile devices to exchange information. However, the intranet provides a private network accessible to a limited amount of users. Traditionally, these private networks often belong to large organizations known as corporations. KITS's intranet site is accessible to only one type of user. In order to access the site, a person has to be an employee of the KSC education department. Through the KITS intranet site, all KSC Office of Education personnel are able to create a record that contains basic information about the intern. The KSC Office of Education needs to manage data related to an intern like the mentor name, the year and term of the internship, office location, etc. This project will focus on assisting the office of education users on the KITS website. The primary goal of this project will be to enhance the site search query using several different categories and searches. Search categories such as: first name, middle initials, last name, home city, home sites, local city, classification, age, gender, ethnicity, major, minor, congressional district, season, year, funding source, number of NASA internships, school name, and school location will be included in the selection menu. A demo

---

<sup>1</sup> Intern, NE-C2, Kennedy Space Center, Florida A&M University.

presentation will be presented to the KSC office of education department, upon completion of the site. The programmer challenges and personal experience while creating this site will be shared in the discussion section of this report. Ruby will be the primary programming language used to build KITS.

## II. Ruby

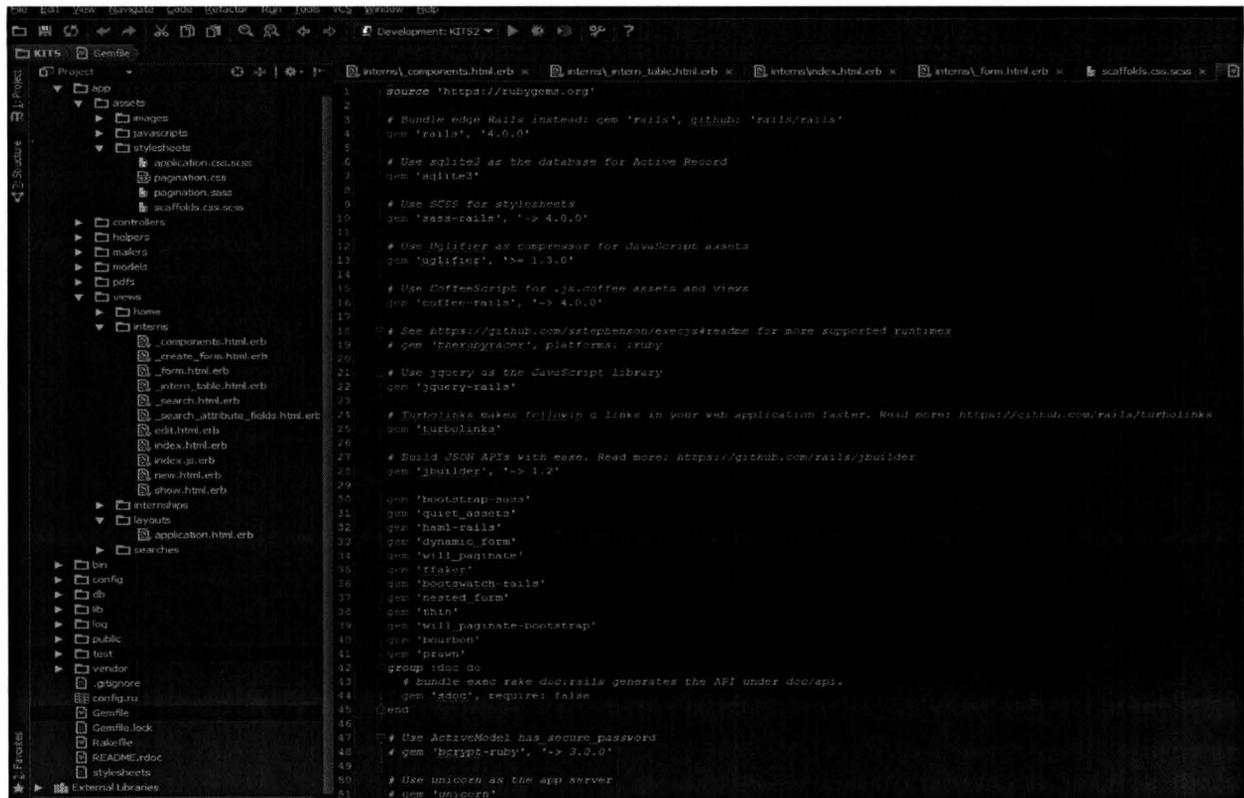
Ruby is a multi-paradigms programming language. It is considered a functional, object oriented and imperative language. Ruby has features similar to Smalltalk and is inspired by Perl syntax. There are also several other programming languages that ruby is inspired by, such as: Eiffel, Python, Lisp, CLU, and Dylan. Many people interpret Ruby as a scripting language because it gives immediate feedback during development, operating systems can be call directly, and it uses regular expressions and powerful strings. In Ruby everything is an object, which is the primary reason why it is viewed as an object oriented language. Other reasons that support the use of the object oriented paradigm in Ruby includes: classes, methods, inheritance, closures, iterators, singleton methods, multiple precision integers, dynamic loading, a large collection of libraries (gems), and exception handling. However, unlike object oriented languages memory management is automatic, variable declarations are not needed, and variables are not categorized in a type. A Japanese Computer Scientist by the name of Yukihiro Matsumoto invented Ruby in 1995. Upon creating Ruby, Matsumoto intended were to create a scripting language that was more object-oriented like than Python and more powerful than Perl. When once asked about why he created Ruby, Matsumoto stated “I hope to see Ruby help every programmer in the world to be productive, and to enjoy programming, and to be happy. That is the primary purpose of Ruby language [4].”

### A. Ruby on Rails

Ruby on Rails (or Rails) is an open source web application framework that is built on the Ruby programming language [1]. The framework has been affectedly improved by software developers around the world, since its first release. There are several reasons why Ruby on Rails is viewed as a convenient tool. One it makes the assumption that there is only one correct way of doing things and defers the developer away from bad alternatives. Two, less code is required versus other frameworks and languages, because it emphasizes the concept of “Don’t Repeat Yourself (DRY).” Three, Rails has the tendency to keep up with the new development trends of the web development world. Four, rails has a large community of developers; that in hence makes it easier to resolve application errors that may seem to be extremely challenging. Finally, all of these reasons enable rails to have a tremendous increase in productivity. Ruby on Rails was developed by a Danish programmer name David Heinemeier Hansson. Hansson extracted the concept of Ruby on Rails from a previous project called Basecamp for a company called 37signals. The first release of Ruby on Rails came in 2004 under the Massachusetts Institute of Technology (MIT) License one. Hansson motive for creating Ruby on Rails was to increase the “programmer happiness” by freeing them from repetitive coding [6].

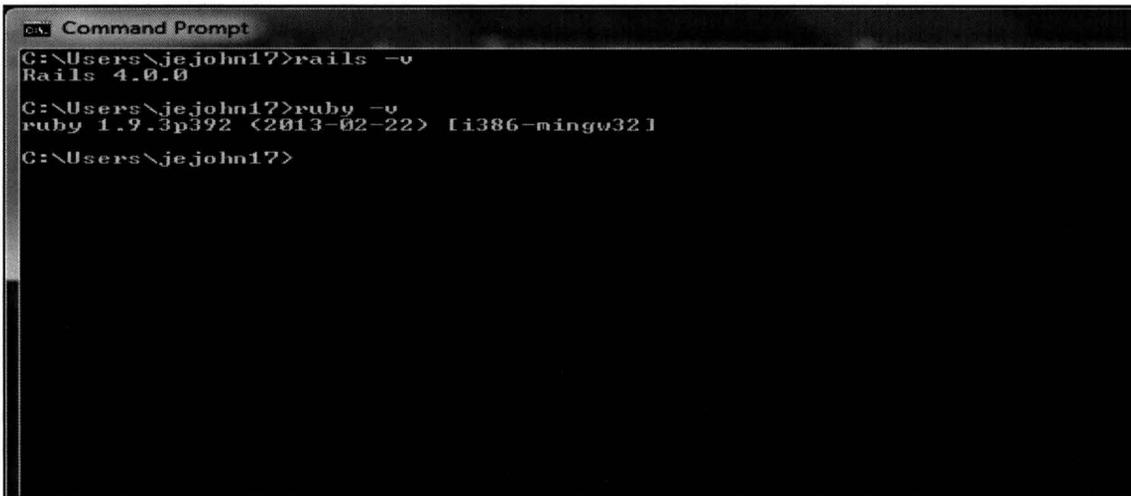
## III. Project Tools

The tools used for this project were RubyMine (version 5.4.2), Rails installer (version 2.2.1), Chrome Browser (version 28.0.1500.72 m), and the Windows Command Prompt (version 6.1.7601). RubyMine is an Integrated Development Environment (IDE) for Ruby and Ruby on Rails [5]. When files are created in a RubyMine project they are stored in their respective path directory on the computer’s hard drive. The project files can also be operated from the hard drive; however, it is preferred to operate them from RubyMine. Some of the files operated out of the RubyMine for this project include: Hyper Text Markup Language (HTML), JavaScript, Cascading Style Sheets (CSS), and Gems (library file), and Database.



**Figure 1. RubyMine v.5.4.2 IDE - On the left side of the screen is a view for the project directory structure folders. The right side of the screen contains an open view of the project Gemfile.**

Rails Installer is an executable file that enables the deployment of Ruby on Rails applications [3]. When installed the latest version of all the necessary files are included in the Rails Installer. The files that come in the Rails installer package can be installed separately, but the installation process may be more tedious [2]. Google chrome was the browser used to test the deployment of the KITS site. The majority of the implementation errors were read from the Chrome browser. Whenever a user hovers over an element in chrome and right clicks on “Inspect elements,” detail information about the element is displayed in HTML and CSS. This chrome feature is known as Chrome DevTools. Finally, all of the commands during the KITS project were run from a command prompt on a windows 7 operating system. Rails commands like “rails new project” and “rails scaffold project item: type” were used to generate files in the project directory.

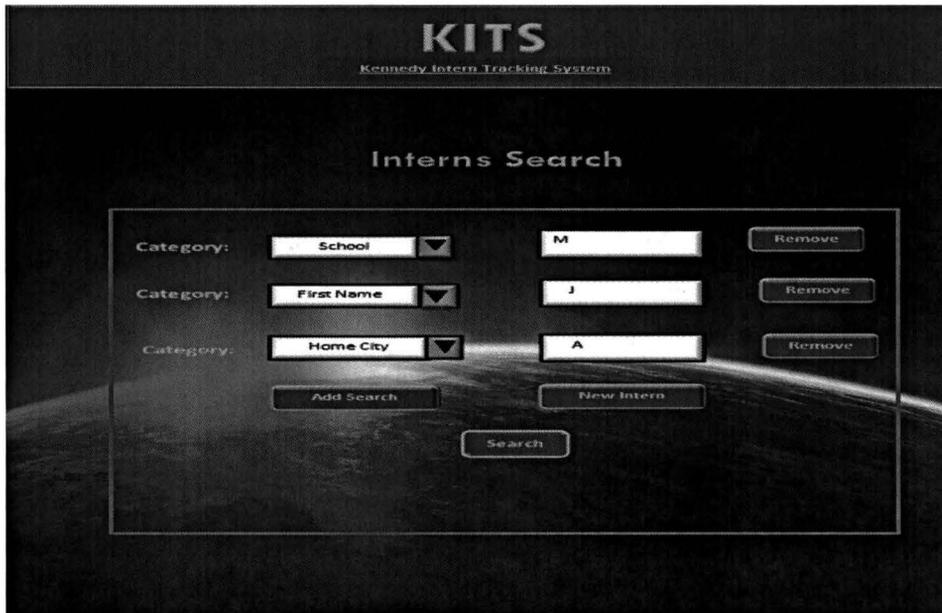


**Figure 2. Windows Command Prompt** – Returns output “Rails 4.0.0” and “Ruby 1.9.3”, after the user enters in commands “rails – v” and “ruby – v” respectively.

#### IV. Scenario Walk Through

**Action:** Interns Search

1. The User chooses the categories for the search and enters in the criteria in the text field. Next, the user clicks the “Search” button.



**Figure 3**

2. Results are rendered from the user’s search.

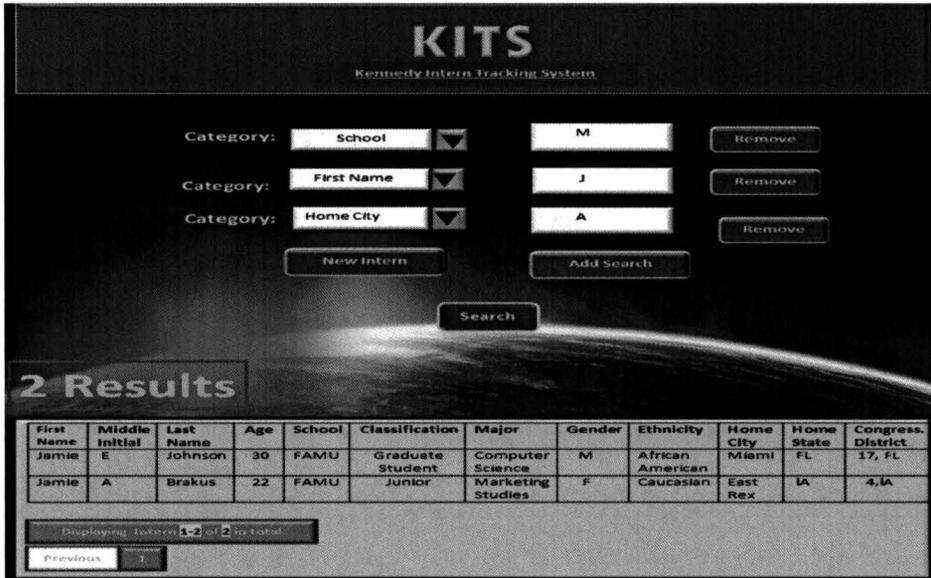


Figure 4

Action: Create a new Intern

1. User clicks on the “New Intern” button.

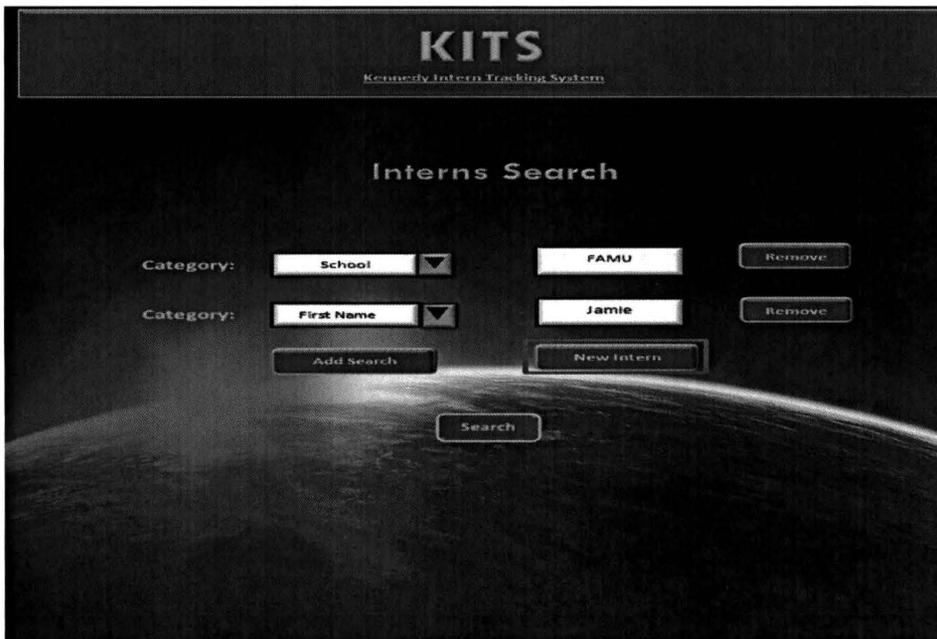


Figure 5

1. The User enters in the new intern information in all the corresponding text fields. Next the user clicks on the “Create Intern” button.

Figure 6

2. The user searches for the new intern by entering the intern’s first and last name. Finally, a result is rendered from the search.

First Name	Middle Initial	Last Name	Age	School	Classification	Major	Gender	Ethnicity	Home City	Home State	Congress. District
Jamie	A	Brakus	22	FAMU	Junior	Marketing Studies	F	Caucasian	East Rex	IA	4,IA

Figure 7

## V. Conclusion and Acknowledgements

Many experts recommend learning Ruby before Rails. Prior to this internship I had no prior experience with Ruby. And although I had prior experience with several other scripting and object oriented languages; I found learning Rails to be a steep learning curve. However I was up for the challenge and after sticking with it I found Ruby to be one of the most powerful programming languages I have come to know. What I enjoy about Ruby, so much is its large collection of libraries known as Gems. In the process of developing the KITS's site I found Ruby Gems to bail me out of almost every difficult situation as far as implementation is concerned. Ruby Gems have the ability to modify and add functionality and features to a website. After using Rails I can now say that it is one of the most productive frameworks I have ever used. All of my communication through Rails was ran using the windows command prompt. Through the command prompt, I keyed in commands that allowed me to migrate (or update) the database, perform unit testing on my code, automatically generate files through a process called Scaffold, and upload my files online. My main challenge was getting a thorough understanding of how everything worked together. When using Ruby on Rails with the RubyMine Integrated Development Environment there are many files to refer to, and trying to figure out how they all connect can be difficult. The strategy I used in dealing with this challenge was for every aspect of the application framework I didn't understand; I took the time to research the concept. I also took notes along the way; therefore, I could refer back quickly to a concept when needed. These techniques of learning not only enable me to complete my project, but get a thorough understanding of every concept I encountered along the way. In addition to learning Ruby on Rails, I was able to enhance my programming skills in JavaScript, JQuery, and Cascading styling as well. These were all supplemented languages that were used along with Ruby to development the KITS site. Yet, what I've enjoyed the most about this internship experience was my support system. I was fortunate enough to have a team of individuals that were patience and committed to guiding me through the difficult intervals of my project. Without them the process in completing this project would have been rigorous. The following are the names of the individuals that assisted me through the completion of a successful assignment: Andrew Davis (mentor), Samuel Goff (NE-C2 Intern), and Jake Hochstadt (of ITC20). Every day of the internship, I had support as far as my project was concern. Andrew was around most of the time, but when he was not Jake and Samuel filled in for him. In closing, my project was a success. I was able to achieve all the specifications and goals my clients wanted me to achieve. In terms of the project evolutions, there are plenty of features that could be added to the KITS prototype I created. Overall I enjoyed my internship experience at KSC. Everyone including: the civil servants, fellow interns, and contractors play their role in making it an enjoyable experience for me. Furthermore, if someone was to ask me "what is the one thing that I can take from this entire experience at KSC?." I would say I became a better debugger in reference to programing languages like Ruby.

### References

- <sup>1</sup> Bigg, R., Katz, Y., Klabnik, S., *Rails 4, Manning Publications*, United States, 2013, Chaps. 1-5.
- <sup>2</sup> Davis, A., “*Taking NASA to the Next Generation Using Ruby on Rails*,” Kettering University, September 2012.
- <sup>3</sup> RailsInstaller, Engine Yard, Ver.4.0.0, San Francisco, CA, 2013.
- <sup>4</sup> “Ruby”, Wikipedia, United States, July 2013. [[http://en.wikipedia.org/wiki/Ruby\\_\(programming\\_language\)](http://en.wikipedia.org/wiki/Ruby_(programming_language))]
- <sup>5</sup> RubyMine, JetBrains, Ver. 5.4.2, Praha 4 – Nusle – 147 00, Czech Republic, 2013.
- <sup>6</sup> “Ruby on Rails”, Wikipedia, United States, July 2013. [[http://en.wikipedia.org/wiki/Ruby\\_on\\_Rails](http://en.wikipedia.org/wiki/Ruby_on_Rails)]