

# First Year Assistant & UCAP I

By

Oliver Hardwick & Rob Scheidler

Submitted to  
The Faculty of the Department of Information Technology  
In Partial Fulfillment of the Requirements for  
The Degree of Bachelor of Science  
In Information Technology

The author grants to the School of Information Technology permission to reproduce and  
distribute copies of this document in whole or in part

Learner's signature: Oliver Hardwick & Rob Scheidler

Date: 4/18/16

Advisor's signature: Rob Scheidler

Date: 4/18/16

University of Cincinnati  
College of Education, Criminal Justice, and Human Services

April 2016

## Table of Contents

| Section               | Page |
|-----------------------|------|
| Abstract .....        | 3    |
| Introduction .....    | 4    |
| Problem .....         | 4    |
| Solution .....        | 5    |
| User Profile .....    | 6    |
| Timeline .....        | 6    |
| Proposed Budget ..... | 8    |
| Testing Plan .....    | 8    |
| Conclusion .....      | 9    |
| References .....      | 11   |

## Figures

|                            |   |
|----------------------------|---|
| Fig. 1, User Profile ..... | 6 |
| Fig. 2, Timeline .....     | 6 |

## **Abstract**

First Year Assistant and UCAPI are dual applications that work together towards two common goals. To provide students with resources from their colleges and help developers with utilizing the same resources for future applications. First Year Application is our showcase of how easily utilization of UCAPI is for developers. UCAPI (University of Cincinnati Application Program Interface) is the aggregation and organization of information of individual colleges into a unified resource.

UCAPI will be available to the public allowing developers to begin testing their application ideas for University of Cincinnati inspired projects. By providing an easy connection to university resources, these applications increase the success of each student at the University of Cincinnati and inherently the university altogether.

## **Introduction**

University of Cincinnati began a program called Learning Communities (LC) out of the department First Year Experience (FYE). While 80% of students that complete the LC program recommend them to incoming Freshmen students it is not possible to bring the opportunities of the program to every Freshman Student. “Most UC students—as many as 75% of incoming baccalaureate students—enroll in a Learning Community or similar educational structure...” (Center for First Year Experience, “FAQ’s”). Leaving 25% of students do not participate in this program which creates the need for better education of becoming a successful student.

Learning Communities provide students with experience in establishing close relationships with professors, explore areas of academic interest, building friendships with classmates, achieving their academic goals, experiencing the feel of a small college with the benefits of a major university, and obtaining reserved seats in popular difficult to-get-into courses. Many of these aspects provided cannot be solved outside of a LC, yet at the same time there are ways to provide a better experience than leaving 25% of students in the dark.

## **Problem**

Learning communities are run by older students for incoming freshmen students to better develop the first year experience at college. The proper education of becoming a better student is especially needed in schools of Cincinnati’s population of undergraduate students. Without this, the possibility of an extreme disconnect between the university and its incoming students allows for major issues for both parties. Learning Communities have proven results in combating these issues for new students with professional development, individual college retention, university engagement, and most importantly improved grade point averages. The First Year Assistant project began with inspiration to help freshmen students become more knowledgeable with University of Cincinnati by providing information in a easy to access unified format. That

project led us to the discovery that data on the University of Cincinnati is owned by individual colleges and access is near impossible to acquire.

With information provided by the University of Cincinnati scattered in different formats and data ownership creating applications for any student is extremely difficult. The majority of this information is hidden by obscure acronyms and that never follow a similar replicable format. General acquisition of this information is only obtainable through different mediums.

## **Solution**

With each college's information individually collected into a unified format. Utilization of the organized data brings together resources that change dynamically by each college. The data collected from public university information is placed into an ease of use public api, otherwise known as the UC-API. The application program interface can be easily adjusted for future changes that may cause inconsistency, but more importantly create an organizational pattern of data showcased through the First Year Assistant. Enabling future developers to utilize resources otherwise not provided to them.

First Year Assistant will connect first year students to necessary resources provided by the university without needing knowledge obtained from other students. Including search functionality of colleges, schools, majors, and resources from our UC-API.

## User Profile

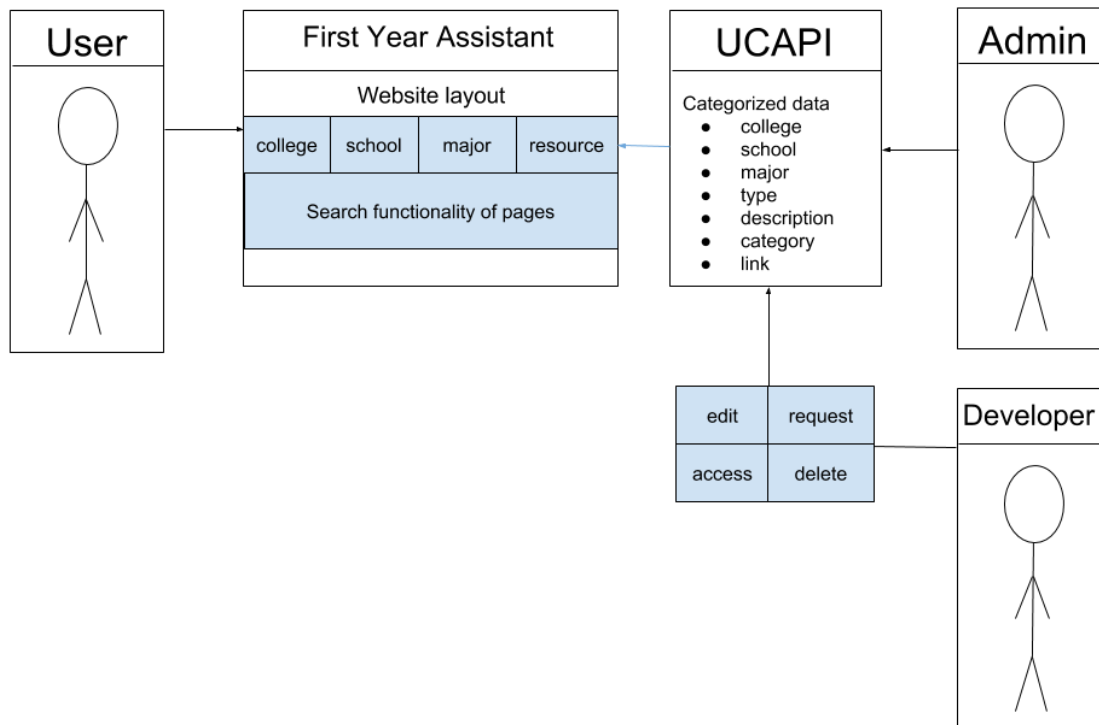


Figure 1, User Profile

## Timeline

| Sprint | Start     | End       | Steps   |
|--------|-----------|-----------|---|
| A      | 9/28/2015 | 11/2/2015 | <b>Find university resources</b>                                  |
|        |           |           | 1 Connect with FYE* to find all available documentation resources |
|        |           |           | 2 Conduct Survey of first year students to find desired resources |
|        |           |           | 3 Assemble rough draft of application layout                      |
|        |           |           | 4 Find all available resources provided by the university         |

| Sprint | Start      | End        | Steps                                 |  |
|--------|------------|------------|---------------------------------------|--|
| B      | 11/2/2015  | 11/23/2015 | Create data layout and data interface |  |
|        |            |            |                                       |  |
|        |            |            | 1                                     | Design data layout   |
|        |            |            | a                                     | Decide best method to distribute data  |
|        |            |            | b                                     | Test   |
|        |            |            | 2                                     | Create data interface  |
|        |            |            | a                                     | Generate method to provide data to application                                       |
|        |            |            | b                                     | Test   |
| B      | 11/2/2015  | 11/23/2015 | Create base website layout            |  |
|        |            |            | 1                                     | Implement Class Structure  |
|        |            |            | a                                     | Design base classes for html/javascript  |
|        |            |            | b                                     | Create speciality classes for each platform  |
|        |            |            | 2                                     | Structure Data Retrieval   |
|        |            |            | a                                     | Consult with data design for best methods  |
|        |            |            | b                                     | Implement classes for data structure of applications                                 |
|        |            |            | c                                     | Test data retrieval methods for shortcomings   |
| C      | 11/23/2015 | 12/14/2015 | Get feedback of current layout        |  |
|        |            |            | 1                                     | Generate Survey of current application   |
|        |            |            | 2                                     | Survey first year students provide feedback of layout and information of application |
|        |            |            | 3                                     | Obtain feedback from FYE on data layout  |
|        |            |            | 4                                     | Prioritize feedback and decide solutions   |
| D      | 12/14/2015 | 2/15/2016  | Improve Application                   |  |
|        |            |            | 1                                     | Adjust layout and information in accordance with provided feedback                   |
|        |            |            | 2                                     | Begin UI/UX design to base application   |
|        |            |            | a                                     | Create format that fits both platforms   |
|        |            |            | b                                     | Adjust base designs accordingly  |

Figure 2, Timeline

## Proposed Budget

Our budget will be offset mainly by using resources currently provided by the University of Cincinnati. We will use the ufilespace to provide a majority of our services provided by the server. Without using the university's servers, I would expect to spend around \$1,000 on a server. Along with this, an internet connection would cost around \$60/month. Also, the price of a domain name would cost around \$15/month. The amount of work involved would be 25-30 hours a week. Charging \$40/hour, would be (\$40 \* hours). Another cost, although miniscule would be the price of power. I would estimate no more \$100/year.

## Testing Plan

In order to successfully complete our application, testing was essential. This section is intended to outline methods used while developing our applications.

### Data Collection

1. Create program to collect information from list of websites that use common template.  
Store data collected in a comma separated value file.
2. View data in Microsoft Excel to ensure data collected is correct.
3. After the all data has been verified to be accurate in file, use Microsoft Excel functions to generate list of insert statements containing collected data.
4. Open command line and connect to database.
5. Copy list of insert statements and paste statements in command line.
6. After all insert statements have ran, print all records inserted.
7. Verify number of records inserted match number of insert statements generated.
8. Visually inspect records printed to ensure all data has been inserted correctly.



## Database Connection

1. Attempt connection to database from nodejs server.
2. If connection is successful:
  - a. Run select statements to collect all information stored on database.
  - b. Generate new javascript files that will store database records as JSON variables.
  - c. Log successful connection in file stored on nodejs server.
3. If connection fails:
  - a. Continue using known data from last successful connection.
  - b. Log unsuccessful attempt to connect to database in file stored on nodejs server.

## First Year Assistant Login

1. Create user on First Year Assistant website.
2. Login into First Year Assistant website using this created user's credentials.
3. Add favorited resource to user.
4. Print user's favorite resources to web console.
5. Visually verify user's favorites are correct.
6. Connect to database.
7. Print all records of user and visually verify all information is accurate.
8. Log off website.
9. Ensure all user information is removed from browser's local storage by attempting to print information to web console.
10. Log back into First Year Assistant website and print user data to web console.
11. Again, visually verify all user information is accurate.

## Conclusion

The value of a University of Cincinnati education is priceless. With this web application, we intend to increase the amount of successful students at the University of Cincinnati. By giving

students this ease of access the actions in improving their education becomes more productive than passive. We found that many of the resources remain scattered and databases we wish to use remain inaccessible at this time. Pressing on with data that does not change as frequently will allow this application to still successfully help students. This will make creating dynamic data within our application more difficult to access, but we see that by changing our current scope away from helping each individual student to the overall student body to be just as necessary. Initially the project began with a mobile application that would provide the resource from within the university ecosystem. As we progressed through our iterations different roadblocks forced us to change our final project goals. Databases we are granted access to and the amount each college wishes to participate led to clear halts in programing. The main reason First Year Assistant became a web application, rather than native mobile application, was the lack of support for OAUTH which would allow students to login to the system. Most of our research is based on speaking with UCIT professionals about how to create an application that does not lose value by April 2016. We took it upon ourselves to collect college information in order to cease the roadblocks we reached for future developers. By creating an easily accessible and organized api other developers will not have the struggles we reached with obtaining information.

## References

"FAQ's." *University of Cincinnati*. Center for First Year Experience, n.d. Web. 25 Sept. 2015. <[https://www.uc.edu/fye/learning\\_communities/first-year/faq\\_s.html](https://www.uc.edu/fye/learning_communities/first-year/faq_s.html)>.

"Learning Communities." *University of Cincinnati*. Center for First Year Experience, n.d. Web. 25 Sept. 2015. <[https://www.uc.edu/fye/learning\\_communities.html](https://www.uc.edu/fye/learning_communities.html)>.