

Form Buster

Milestone 6 Progress Evaluation

Team Members

Grace Dolphy - gdolphy2015@my.fit.edu

Raphael Setin - rsetin2015@my.fit.edu

McNels Sylvestre - msylvestreph2016@my.fit.edu

Faculty Sponsor

Dr. Eraldo Ribeiro - eribeiro@cs.fit.edu

Client

Dr. Philip Chan - Computer Engineering and Sciences at FIT - pkc@cs.fit.edu

Meeting(s) with Faculty Sponsor

04/15/19 1:10 PM - 1:20 PM

Meeting(s) with Client

03/29/19 4:00 PM - 4:30 PM

04/09/19 2:50 PM - 4:10 PM

04/10/19 4:10 PM - 4:40 PM

Progress of Current Milestone

Task	Completion %	Raphael	Grace	McNels	To do
1. User manual	100%	33%	33%	33%	None
2. Demo video	100%	33%	33%	33%	None
3. Conduct user experience survey	100%	33%	33%	33%	None
4. Complete Co-Prerequisite waiver form	100%	33%	33%	33%	None
5. Form due date	100%	33%	33%	33%	None

Discussion of each accomplished task (and obstacles) for the current Milestone

- ❖ **Task 1:** A manual is useful in that it contains instructions and respective diagrams on how to properly use the system, for all users. It also has use cases to facilitate comprehension. The content of the manual is divided based on the user type. Each section thus presents the features available to a particular user type. For each feature, the execution steps are outlined at first, and screenshots of the website pages relevant to the feature are displayed, labeled with the steps numbers.

- ❖ **Task 2:** The demo video's goal is to show the main features of the system, similar to commercials for a certain product. The first part of the video shows that a student can submit forms, get forms approved, and receive notifications. Then, it goes on to talk about the features available to faculty members such as approve/decline forms, student look up and history of signed forms. The features for staff members are presented in similar fashion, and it ends with capabilities offered to student coordinators.

- ❖ **Task 3:** This survey provided insight from the users' perspective. Its results were displayed on the showcase poster. It was an opportunity to get an outside opinion and to see where the system can improve. Users were asked to do three tasks pertaining to their role, without any specific instructions on how to execute them. The goal was to test how intuitive the interface was to the users. After execution of the tasks, they were asked to complete a survey regarding the difficulty of the tasks, how they would rate the system and if they would recommend FIT to use it over the current system. The results revealed a great level of satisfaction as 88.4% of the surveyed users said Form Buster was better compared to the current system and 94.1% of them said they would recommend Form Buster over the current system.

- ❖ **Task 4:** This task required refactoring of code, and generalizing of code in order to allow both the registration form and the new co/prerequisite form to use similar functions that have already been used for the registration form. For this new form, the user will select the term they would like to register for, and add one course from the results that they would like to add to the Co/Prerequisite form. We have implemented the algorithm to allow the user to search for a co/prerequisite course which will be independent of the semester, since we are not aware of an FIT courses api that has all of the courses that the school offers, we queried the courses api with each fall, summer, and spring and obtained the results, only printing course titles, not including any duplicates. If a course was

not listed in the fall, summer or spring, we assumed that FIT does not offer this course. It caused difficulties trying to implement this algorithm as the results from the three queries did not return in time before it could be used, so this required some refactoring of the code. Like the original co/prerequisite form, the user will only be able to add up to 2 courses to waive. The text field for the last section of the form was added in so the user could add a waiver justification reason. Now when a user submits the form, all values are added into the database for later retrieval. The new submitted form shows up in the students dashboard, the student is able to view the form, view the tracking of the form (which now has the Head of the Department approval waiting). The advisor, head of department and staff can mark the form as approved/declined and the staff can mark as processed/not processed.

- ❖ **Task 5:** Forms are now sorted in ascending order for the “In-Progress Forms” page of Students; descending order for the “My Completed Forms” page of Students, Faculty, and Staff; descending order for Completed forms while viewing from the Student Coordinator perspective; and ascending order for In-Progress forms from the Student Coordinator perspective. Faculty and Staff now have their “Pending Forms” page to be sorted in ascending order for due dates, and if the due dates are the same, then ascending order for submission dates.

Discussion of contribution of each team member to the current Milestone

- ❖ **Grace:** Developed the algorithm to get back a list of all classes that FIT offers, independent of term; the FIT courses API was used, where each term was queried with the user’s search query for the new co/prerequisite form. Finished up the rest of the general UI for the co/prerequisite form, and allowed the student to start the form, save the form for later, adding this information into our Form Buster database, or choose to submit the form then adding this information into the database as well. Enabled the student coordinator to be able to start the co/prerequisite form and send it to the student for editing. Developed the draft viewing and editing for the new form page.
- ❖ **Raphael:** Developed the entire tracking bar for the approvals; made the buttons and the titles of the pages to be more action centric, like changing “Dashboard” for students to “In-Progress Forms”; added a totalCredits field to all the student users in the pseudoPAWS database, so we would be able to compute due dates based on their academic standing; implemented the entire due date from scratch,

in a way that Faculty and Staff would have 3 business days to approve/process their pending forms; implemented the sorting for all the user's forms, including drafts -- now Faculty and Staff members have their forms sorted out by due date, and if the due dates are the same, then it is sorted by submission date (oldest first); added form deadlines for students and student coordinators, both inside the forms, as well as when starting a form (dynamic deadline); adjusted the "Look for Students Records" page for Staff, Faculty, and Student Coordinators, to make it show the student info first (including their total credits and their academic standing) before actually showing their forms; added a term field to the forms and also limited the "Start a Form" process depending if the form is available for a term; added a feature to automatically delete drafts in which are for older terms; added the dates in which a form is available to start/end in the "Start a Form" page; added missing CEU information in all the forms (wasn't working before); designed the entire login page from scratch, with added recaptcha in case user enter wrong password too many times; corrected the authentication and now everything works flawless -- people cannot access the user pages anymore by just typing the URL; corrected many bugs from the drafts page and some other pages in the website, as well as refactored some code too; blocked the student from submitting a form if the form was incomplete, both in Drafts as well as when starting a new form; improved the whole website aesthetically to make everything look better and more modern than before; removed unfinished features like "Enter a PIN" for Faculty, "Forms Management" page for Staff, search bar in "Start a Form" page, etc.

- ❖ **McNels:** Wrote the user manual. Drafted questions for the user experience survey and conducted it with potential users of the website. Directed the production of the demo video for the website.

Lessons learned

- ❖ Lesson 1:
Asynchronous behavior and promises in Javascript. We had many problems in the beginning of the project with this feature of Javascript, but we ended up mastering it at the end of milestone 5, where all of us understood how it worked.
- ❖ Lesson 2:
Communicate with the client as much as possible to define priorities of the project. By not communicating too much at some instances, we ended up

developing a feature in the wrong way and thus we had to change it later to achieve the client's desires.

❖ Lesson 3:

Spend time doing general research about a tool instead of figuring out how to use it for something specific with no prior knowledge. We wasted a lot of time trying to learn on the go without having researched first; this ended up being clear at the end of milestone 5 and milestone 6.

Sponsor feedback on each task for the current Milestone

❖ Create user manual:

Great.

❖ Demo video:

Great.

❖ Conduct user experience survey:

Great.

❖ Complete Co/Prerequisite form:

Great.

❖ Form due date:

Great.

Sponsor Signature: _____ Date: _____

Sponsor Evaluation

Sponsor: detach and return this page to Dr. Chan (HC 322)

Score (0-10) for each member: circle a score (or circle two adjacent scores for .25 or write down a real number between 0 and 10)

Grace	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Raphael	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
McNels	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10

Sponsor Signature: _____ Date: _____