

# Form Buster

## Milestone 3 Progress Evaluation

### Team Members

Grace Dolphy - [gdolphy2015@my.fit.edu](mailto:gdolphy2015@my.fit.edu)

Raphael Setin - [rsetin2015@my.fit.edu](mailto:rsetin2015@my.fit.edu)

McNels Sylvestre - [msylvestreph2016@my.fit.edu](mailto:msylvestreph2016@my.fit.edu)

### Faculty Sponsor

Dr. Eraldo Ribeiro - [eribeiro@cs.fit.edu](mailto:eribeiro@cs.fit.edu)

### Client

Dr. Philip Chan - Computer Engineering and Sciences at FIT - [pkc@cs.fit.edu](mailto:pkc@cs.fit.edu)

### Meeting(s) with Faculty Sponsor

11/26/18 (10:00 AM - 10:50 AM)

### Meeting(s) with Client

11/09/18 (5:00 PM - 6:00 PM)

### Progress of Current Milestone

Task	Completion %	Raphael	Grace	McNels	To do
1. Implement, test, and demo digital approval for Students and Faculty	60%	20%	20%	20%	Finish approvals
2. Implement, test, and demo database lookup for student records	100%	33.3%	33.3%	33.3%	None
3. Implement, test, and demo email notifications feature	60%	20%	20%	20%	Write listeners for notifications
4. Implement, test, and demo form tracking feature	100%	33.3%	33.3%	33.3%	None

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

- ❖ **Task 1:** A temporary Start a Form feature was implemented from the students side for the registration form as planned. We decided not to use a pre-filled in form to populate the database, but instead we implemented the form fields so the user could fill out and submit the form, and the data would then be extracted from the form to be shown to other users. Lastly, the approval wasn't finished yet, because we had to spend time on refactoring the whole project's code; but we have the function to dynamically visualize the forms already, so it's just a matter of having a checkmark and a text input besides the Faculty or Staff that actually needs to approve that form, and send that approval to the database.
- ❖ **Task 2:** Each Faculty, Student Coordinator, and Staff have a "student's page", where they can view a list of all the students in our FormBuilder database. When a Faculty, Student Coordinator, or Staff member selects a student from the list, they will be able to view a student's completed forms and in-progress forms, which come from our Formbuster database.
- ❖ **Task 3:** The code to create the notification information in the database has been implemented. Sending emails from the notifications created in the database requires implementing a listener for said notifications. Once a notification created, the listener relays the information to the email delivery system, which in turn sends the notification out to the appropriate user. The medium used for that purpose is Sendgrid as it allows using Firebase Cloud Messaging, which permits writing such listening scripts. The setup for Sendgrid has been completed at this point, however the listening scripts need to be completed for each notification event. For example, once an approval is obtained for a form, a notification object is created in Form Buster's database. There is a script in Firebase Cloud Messaging that listens for the creation of such objects. The script informs Sendgrid that an approval notification was created and Sendgrid sends the template email that was created during setup to the student.
- ❖ **Task 4:** Whenever the Faculty or Staff approves or declines a form, the Student will be able to see the changes in his view, by refreshing the page. If they declined the form, the student will also be able to see the reason why they declined. The approval's section of every form will be available for the student, faculty, student coordinator, and staff, when they search for a student's form.

## Discussion of contribution of each team member to the current Milestone

- ❖ **Grace:** Worked on the start a form feature, where a user may fill out the Registration Form and submit or save the form, as well as the confirmation page. The registration form allows the user to add and delete rows for entering course information. The form also checks to make sure that the user is submitting at least 1 row, but no more than 8 rows. Added the form information into the database, so that it will appear in the student's dashboard. Programmed the in-app notifications logic, which reads from the database and populates the user's view with their notifications, and used moment.js to provide relative times from when the notification was created (i.e 7 mins ago). Also programmed the creation of an in-app notification, which is created in the database. Made the student details page which is shown when a faculty member selects a student member, it reads from our pseudo-PAWS database, to get the student's information, and also leads to the student's start a form feature.
- ❖ **Raphael:** Provided a nice way to see information in the forms, through a tooltip library called Tooltipster. Used that to show, for example, the information of who has signed a form and who has not. Additionally, implemented a function to view all the information of a form (in a popup), once the user clicks on the form entry. Implemented the student search page, in which faculty, student coordinators, and staff can search for a student with either their first name, last name or 900 number; finished the implementation of searching a student by a full match of first name and last name. Also, implemented the visualization of completed forms in-progress forms of a student, when the faculty, student coordinator, or staff wants to see the student's records. Implemented the popups of the Faculty -- *where they can see the student's information and click on "Generate PIN", "View Records", or "Start Form" buttons* -- and Staff -- *where they can see the student's information and click on "View Records" button*. Lastly, refactored all the code for the whole project in order to make everything more understandable, and fixed many tiny issues that the project had, of both front-end and back-end.
- ❖ **McNels:** Worked on an alternative authentication approach to overcome session persistence issues across subdomains when using Firebase Auth. The best fix was to use a third party AaaS (Authentication as a Service) provider in coordination with the existing user database in Firebase. Explored Auth0 and Okta as potential providers. Chose Okta over Auth0 as it allows more customization when it comes to setting up the system and where the system is hosted. At this moment, only the linking to the existing database and the redirection to the respective homepage is left for the Okta authentication system

to be complete. Worked on the email notifications feature to alert users of an event that is pertinent to them. Investigated email automation platforms and decided on using Sendgrid as opposed to building a email delivery server from scratch.

### Plan for the next Milestone

Task	Raphael	Grace	McNels
1. Implement pending forms (Dashboard page) for Faculty and Staff	33.3%	33.3%	33.3%
2. Implement the History page for Faculty	33.3%	33.3%	33.3%
3. Implement the Drafts page for Student	33.3%	33.3%	33.3%
4. Finish the notifications (both in-app and email)	33.3%	33.3%	33.3%
5. Finish digital approvals of forms for Faculty and Staff	33.3%	33.3%	33.3%

### Discussion of each planned task for the next Milestone

- ❖ Implement, test, and demo displaying pending forms for Faculty and Staff:  
When a student submits a form, the form will need to appear in the proper faculty's dashboard, and when the form has received all of its approvals, it will need to appear in the appropriate staff member's dashboard. The original form will be saved in the student's database, and we will save references of the original into the appropriate faculty member(s) and staff when it is time for them to sign or approve the form.
- ❖ Implement, test, and demo the History page for Faculty:  
For this, we will need to save all forms that were declined or approved by the faculty into their history. The database should be updated as soon as the faculty has made a decision This information will later be obtained from the appropriate faculty member's database to be displayed in the history page for the appropriate faculty member.
- ❖ Implement, test, and demo the Drafts page for Student:  
For this feature, we will allow a student to save a form when they are in the middle of starting the form. The form should appear under the draft's section of

the student's account, and the database should contain the information that was inputted at the time of saving for later viewing.

❖ Finish the notifications (both in-app and email):

To achieve this, we will have to send approve a form from a Faculty view and a Staff view, and check to see if the Student has received the notification in the email and in his Dashboard. In order to test this feature, all three of us will have to use our own emails in order to test if the email was successfully received or not, since the emails that are currently in our database are not legitimate, thus preventing us from using them to test.

❖ Finish digital approvals of forms for Faculty and Staff:

To achieve this, we will need to complete milestone 1 first, then we will use the form view that we have already to allow the faculty and staff members to view the form, and select approve or decline (and provide a reason for decline). We will need to make sure our notifications, in-app and email, are sent out as a result of the faculty and staff member's form decisions (approve/decline/processed), and that the database is updated properly to know who has signed so far.

**Sponsor feedback on each task for the current Milestone**

❖ Implement, test, and demo digital approval for Students and Faculty:

I understand that you guys haven't finished this yet, but at least it shouldn't be that hard since you already have the visualization of the form working properly. I hope you have it fully working for next milestone, as that is very important.

❖ Implement, test, and demo database lookup for student records:

Everything is looking great, good job!

❖ Implement, test, and demo email notifications feature:

I know that email notifications is hard, but I am happy to see that you did a lot of research on it and now you have better knowledge on how to implement for the next milestone. I am also happy to see that you started working on the in-app notification feature, though it wasn't planned to be implemented in this milestone. Just try to get the email notifications working for the next milestone then.

❖ Implement, test, and demo form tracking feature:

Everything is looking great, good job!

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: \_\_\_\_\_