Development of Osiris App: “Finish all, finish on time”. Improving time management

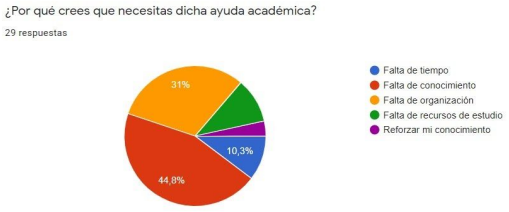
Autonomous University of Yucatán. Fernan Enrique Cetina Escalante, Jorge Teodoro Dawn Rodriguez, Rodrigo Alejandro Castrejón Cervantes & Cinthia January Huchin Pedrero.

Abstract— The research team encountered a social problem within the group: “UADY Software Engineering: first semester students”, the research suggested that time management was an important issue to solve, that’s how the project of creating an app along with users began. This article presents the different stages throughout the development process of the Osiris App. The article mainly includes the recollection of requirements and the design stage where it demonstrates the traceability between the information collected and the high-fidelity prototype. As the development got carried out, the research team encountered that it wouldn’t be possible by technical and time resources to fulfil the needs of users of creating an app, that’s why the efforts shifted towards a high fidelity prototype before a functional app to guarantee acceptability tests and quality controls from users.

Keywords— students, high-fidelity prototype, time management.

I. OBJECTIVES & JUSTIFICATION

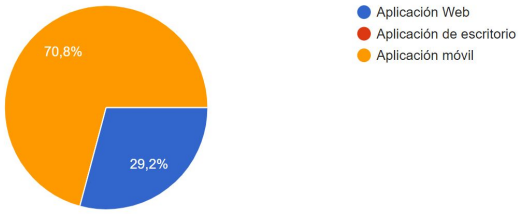
The project began by reaching out to our community: first-semester students of Software engineering of UADY, in order to find problems related to academic progress. After several surveys sent to all our classmates, we discovered that time management was a dominant problem among students.



(From to to bottom: Why do you need such academic help?, 29 answers: Lack of time, lack of knowledge, lack of management, lack of studying material, enhance my knowledge)

The picture above shows the results of said survey. As seen, 31% of them said they needed help with organization, and 10.3% said they needed help with their lack of time. Those two areas combined almost match the one where people answered that they needed help with their lack of knowledge. But we decided to focus on the former because knowledge is gained as time passes, while time and task management depend on the student’s habits. Also, poor time management can negatively impact someone's ability to learn, like when they miss classes or forget to turn in homework.

Following these results we asked them for an efficient way in which we could deliver a solution, that’s how the final purpose landed on creating a mobile app that helps students manage their time and as ‘added value’ we wanted to implement a smart/algorithmic based tool that helps them reschedule their academic and personal objectives passed deadlines.



Translation: 24 answers. WebApp(blue); Mobile App(yellow); Desktop App (red)

Our project intends to help students manage their time, keep track of their future activities and allow them to retrieve time missed. For this purpose, we will develop a tool that allows its users to add and remove tasks. It will also have several options to keep track of them with constant notifications, progress updates and a tool that helps them find a path to complete those activities they missed without overlapping with new tasks.

II. Methodology

Scrum is the methodology that is being implemented due to its flexibility and its ability to adapt to the project. As the framework required roles were assigned, Fernan Cetina being the Scrum master and Jorge Dawn, Rodrigo Castrejón and Cinthia like developers.

The complete explanation on how we adapted Scrum to fit our needs can be seen in the "project follow up" section.

Requirements phase summary

During the first increment we concluded the idea on how we expected the system to perform, the basics that should follow and its restraints. Initially we recollected information of the users from surveys and figured out the objective of our application, after this we began the eliciting process. We were constantly evaluating these requirements, for this we performed another survey and loop in the refinement of the requirements. Then we move forward to the construction of the use cases where we explicitly define the specifications of each requirement, the actions, the exceptions and preconditions. You can check all the information in a more detailed way in the GitHub repository. (link:https://github.com/FernanCetinaE/TeamOsiris)

Design phase summary

During the design phase we used Figma for the Wireframe modeling. Here we prototyped the UI and guaranteed all the traceability aspects to ensure high quality on static and some dynamic testing.
It can be seen here: https://bit.ly/3rizTGe
The main goal of Team Osiris is to provide itself or another team of software development enough tools to guide their own project with the specified requirements. As such it is not displayed: Architectural, Data Types and Procedural design, because it is not intended to show that level of specificity, each team can decide what fits best on future technologies.

Testing summary

For the project, three types of tests were carried out: inspection, 'acceptance testing' and 'five seconds test'. We applied these types for the artifacts released during the design. The first one being inspection, where we, with the help of our mentor, decided if each requirement is properly represented in the design. The 'acceptance testing', where we showed 6 random students our wireframe design and later asked them relevant questions that were intended to help the development of a new version of the project. The final type of testing was the 'five seconds test', where information was

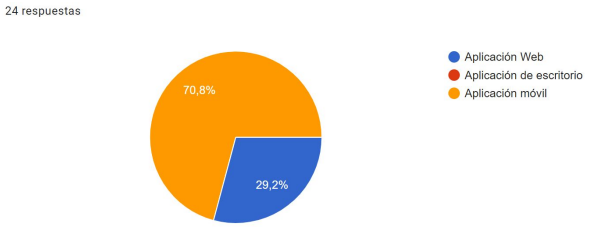
gathered and imported into a spreadsheet, this is intended to provide feedback of the UI design.

We also performed tests in the implementation phase, in this case we proved our high-fidelity prototype. In this phase we also performed a 'five seconds test' with the help of UsabilityHub which allowed us to standardize and finish in a much easier way all the tests. For this test we interviewed 6 students, they commented that it was easy to understand and could identify the different functionalities presented. For the 'acceptance testing' which was applied to 10 random students, we asked them if the interface was intuitive and the users said it was and they could make use of all the functionalities. The students commented that they liked the colors because they are nice and simple. Users were able to identify the buttons without problems, as well as understand the use of the prototype. In very general terms users were satisfied with the prototype.

Implementation summary

For the Minimum Viable Product(MVP) of the third increment the development team made the commitment of releasing a high-fidelity prototype as the Implementation artifact and not the generation of code, because of schedule considerations and the technological constraints. But on behalf of the future we gathered the following information:

For the platform of Osiris we presented the customers with a new survey, in which 24 participants were registered and decided 'Mobile App' for the platform; for this reason we thought that the prototype would be helpful to make instant updates and guide the UI for the Mobile App, that's another reason why we focus on making a functional prototype of the product.



Translation: 24 answers. WebApp(blue); Mobile App(yellow); Desktop App (red)

In this way we focus on developing the high-fidelity product. We performed two work-sessions where we refined the artifacts obtained: the wireframe and the first released sample. The implementation followed a continuous integration methodology where we create the root component and each member works with separate components and merges them to the system, all this process is in our GitHub repository.

III. PROJECT FOLLOW UP

For the 'process' we guided the development phases according to Scrum but as mentioned before we made changes so we could work in a more efficient way. Those changes will be described in this section, along with other "follow up" related topics.

For the Sprint Backlog, there were times in which it was the same as the Product Backlog and the Scrum master on a Sprint Planning stage decided which tasks to work with. For the assignment of activities, we were using Trello, in which every 2 weeks the activities that we must present to the Scrum master were updated and they might be have been daily or when more convenient (this is the equivalent to daily scrums); once an activity is finished, the testing is carried out and we proceed to identify the errors to be corrected, this process is repeated until the

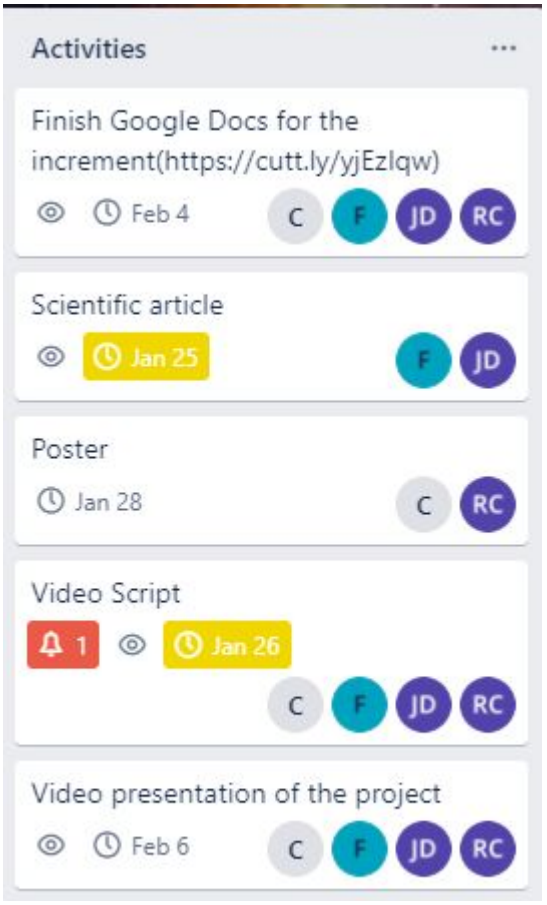
activity complies with our quality control, then the merges are carried out in the GitHub repository.
As a further explanation, a Sprint should have lasted 2 weeks at most.
A day later of the release of an increment a Sprint Retro should have been performed under an hour.

Every aspect of this project can be seen in full lenght and detail on the team's Github repository. The URL is: github.com/FernanCetinaE/TeamOsiris
During the development process we used several tools to help us be more efficient and ease the generation of evidence of the work performed.

We used Microsoft Teams to schedule meetings within the group and to receive mentorship from professor Edgar Cambranes. As an example:

(Translation: Fernan Enrique Cetina Escalante scheduled a meeting Th. 12/17/20)

In addition, every task was scheduled on Trello, assigned with its accountable and due date. As an example:



And once every meeting is counted, tasks completed and the final presentation arrives, it's time to establish the individual contribution of team members in regards to the development process. That's why we worked with the following aspects and formula to measure it.

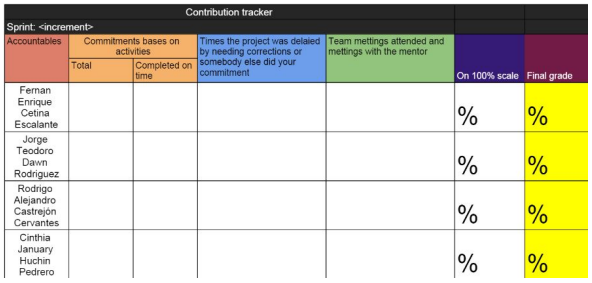
Commitments bases on activities		Times the project was delaied by needing corrections or somebody else did your commitment	Team mettings attended and mettings with the mentor
Total	Completed on time		

[(Activities completed in time/Total activities + your attendance/highest attendance)/2 - if(delay>3){delay*5%} else {without sanction}] * 20%

At the end of the sprint, each member was expected to contribute 20% in total. In the brackets, its shown contribution in base 100%. On a brief note, at the beginning we were 5 team members, that's why initially we expected to contribute on the same

percentage (20%); we later changed it to be a factor of 25%.

As an example:



Contribution tracker						
Sprint: <increment>						
Accountables	Commitments bases on activities		Times the project was delaied by needing corrections or somebody else did your commitment	Team mettings attended and mettings with the mentor	On 100% scale	Final grade
	Total	Completed on time				
Fernan Enrique Cetina Escalante					%	%
Jorge Teodoro Dawn Rodriguez					%	%
Rodrigo Alejandro Castrejón Cervantes					%	%
Cinthia January Huchin Pedrero					%	%

At the end of the third increment we finally filled the contribution tracker of the whole process, it includes the sumatory of all the activities and meetings-assistance of each member with the percentage of their final contribution.

Contribution tracker					
Total contribution					
Accountables	Commitments based on activities		Team mettings attended and mettings with the mentor	On 100% scale	Final grade
	Total	Completed on time			
Fernan Enrique Cetina Escalante	22	22	13	100%	25%
Jorge Teodoro Dawn Rodriguez	21	21	12	96%	24%
Rodrigo Alejandro Castrejón Cervantes	18	18	13	100%	25%
Cinthia January Huchin Pedrero	18	18	13	100%	25%

IV. LESSONS LEARNED

1. Throughout the whole project different tasks like “Work process” or “Meetings” require a different ICT tool, like meetings that are more accessible via TEAMS, Excel tracks better a checklist of contributions or Trello lets us visualize in a graphic way the pending activities.

2. We take into account activities as: requirements eliciting, approving requirements, communication, scrum framework. In a broader perspective, we try to evaluate the feasibility of our project in question as “How are we going to develop this product?”, “Does there already exist another product that provides this service?”.

3. Having a well-organized list of activities so that we could be prepared for each one and advance using our most developed abilities. We took important decisions as a team like prioritize phases. For each of them we achieved a more detailed knowledge of the phases in which we were working.

4. We considered dividing most of the activities and hash out the possible contribution of each member of the team depending on its abilities. At the beginning of any increment each member makes a commitment based on their time and abilities (“Work process”), so being self aware takes great relevance to make a responsible decision that affects your work and others.

5. We learned to interact with users via interviews and surveys, analyze the information collected and elicit requirements the best as possible in order to ensure the UX.

6. As the development of the increment evolved we noticed that complex task as quality assurance, design and implementation (functional code) really require specified professionals with experience in order to ensure that users get the best possible product; we worked our way out by improvising but dedicated professionals in areas like Coding, Quality and UX are key to the success of a project.

7. During the second increment process we needed to learn new abilities and adapt us to the requirements for the development phases involved; in order to reach our goals we had to be self-aware of our edges and organize our time in order to have enough

time for managing our lack of knowledge and our personal lives.

8. Throughout the increments, we spent part of our time learning how to use design tools and trying to use them in the best way. In these cases we worked with Figma and learned the process for refining a prototype and passing from a wireframe to a high-fidelity prototype alongside with users' acceptance.

V. CONCLUSION

As a team we are very pleased with the final result of the project, the collaboration of users and the guidance provided by the mentor.

One key aspect of this college project was to bring first-semester students closer to the job commonly performed by software engineers; and putting the theory in practice and figuring out what fit our needs and decide what didn't, really helped us understand what our career paths looks like: an opportunity to be creative and use what others have found along the way to benefit users.

For the course we think that a lower amount of theory could be implemented because a lot of it is really shifted on how things were done in the past and very little on current examples. Instead, that time can be used on: smaller projects; more talks from other software engineers or other IT fields; analysing how our job has helped industries or something as important as helping students develop soft skills, which can improve their relationships with themselves and society in general.