

Resident Assistant Accessibility Application

Computer Science Capstone

COM-497

Maxwell Kizewski

Jared DeWinne

Ryan Roper

Dr. Manh Van Nguyen

April 23, 2020

Abstract

The Office of Residence Life at Saint Leo University currently has a resident assistant resource website that hosts different quick links and forms that are essential for the Resident Assistant role. Overall, the site has different components and functions, which are functional, but not sufficient in terms of accessibility and security. The site is currently cluttered and is in desperate need of a cosmetic change to appear more well-ordered and eye-catching. Throughout this semester, our group was tasked with using different coding platforms, working with TI3 and the Office of Residence Life at Saint Leo University to design some upgrades and functions that can be added and modified to the site. As a result, it is our belief that we have created a more efficient and purposeful application that can be utilized by Residence Life staff at our university.

Introduction

The residence assistant resource site is currently outdated, unprotected, and underutilized. Opening the site is bland and does not have very much color and or personality to it. Adding on the fact that the site can be accessed by anyone using the internet. This is potentially hazardous to personal privacy as there are resident assistants and professional staff's cell phone numbers, web forms, and other important information held on the site that could cause disruptions to office proceedings. Resident assistants theoretically are supposed to use this site to make their lives easier and offer better access to important information when attempting to do their daily, weekly, and or monthly tasks, but unfortunately the current site is lacking different components to make the site more useful for the Resident Assistant role.

Our mission was to create a new resource website was created to help Residence Life staff access information quickly and offering better mobility by being able to access information on smartphone and desktop platforms. The intent of creating this web application was to give Resident Assistants and Resident Assistant Directors no excuses when it came to knowing and accessing information dealing with the responsibilities of the role. By making information clearly accessible and with Residence Life staff in mind, we were able to embark on a development project that would make the lives of these employees easier.

The current resident assistant resource site has multiple links for different areas of the job. There is a link for duty calendar, bulletin boards to go, and other creative help links that are spastically placed all over. There are also links to several external sites that Resident Assistants use in their day-to-day operations. JAMS is also a link within the website, JAMs is a platform where resident assistants and resident assistant directors have access to write incident reports,

and or track down residents on the campus. Sites like these were not of our concern as these sites are password protected by using the same credentials tied to their corresponding Oktas accounts. This was the inspiration behind password protecting our application and working with TI3 to pull these credentials.

Overall, there are currently a great deal of flaws in the current website that is being used by the Office of Residence Life. Therefore, an auxiliary site that can replicate the same functions, while also introducing new ones would be helpful to Saint Leo's Residence Life team.

Preparation & Brainstorming

As we stated in our introduction, we wanted to create an application that would keep many of the same functions that are currently used in the website while also introducing new features that our Resident Assistants can use to help make their lives easier. Our overall project was initially displayed in our project proposal, which can be seen in Appendix A. Our project proposal was an ambitious one, with the intent of making our new web application a site that would coincide with the existing resource site and would offer many different features. We wanted to create a site that would result in centralizing: the Current RA Website, Duty Log Calendar, Program Calendar, JAMS/Simplicity. New features that could potentially be added include: Instant Duty Swap, Duty Check-in, Directory Access and On-call Quick Access. With these tools implemented into a new environment and centrally located, we felt that we would be able to create a successful application that all of our Resident Assistants would appreciate.

We wanted to focus on a few major tools that would be used by the Residence Life staff on a regular basis. As stated in our proposal, the Instant Duty Swap, Duty Check-in, Directory Access and On-Call Quick Access were essential features that we wanted to try to integrate into

the environment. This proved to be very ambitious and some of these features would eventually be phased out due to time restraints.

First, duty swapping is a helpful tool on the resource site to help resident assistants in a time of need when they need to trade duties with another employee. Currently, a simple duty swap form must be completed if there is a need for one. How the current duty swap form works is that both of the people willing to exchange with each other, must visit the resource site and fill out the swap form. After filling out the duty swap form, it must then be overseen by a professional staff member who will make sure the information is correct. The professional staff member reviewing this form may check that the resident assistants will not be doubled booked and make sure the switch is being completed within the 48 hours in advance of the closest to date duty. Once the form has been overseen and seen as correct, the professional staff member may then make the informed changes on the duty calendar to accurately represent the new information regarding the switch. The hope was to create an application that would automate this process and make the changes to the calendar without intervention of professional staff members.

Duty Check-In was an essential feature that our group wanted to streamline to ensure Resident Assistants are being held accountable for their duty check-ins. Currently, Resident Assistants must check-in for duty via texting into the Resident Assistant Director on-call for that week and notify them where they are working from. However, some Resident Assistants use various different platforms such as Snapchat, WhatsApp, and Facebook Messenger to check-in, which causes complications when the on-call personnel is trying to see who checked in for duty that night. To streamline this process, we wanted to dedicate a division of our application that will automatically pull the name of the account logged into and send that to the Resident

Assistant Director On-Call. This would then have a time stamped check-in in the RAD's Outlook email account. Overall, this will be a viable solution to help provide a steady means of communication that Resident Assistants checking in with the Resident Assistant Director on-call rather than text, which at times can be unreliable due to sparse signals in specific resident halls duty areas.

As mentioned before, currently there is a directory on the resident assistant resource site. This site holds vital information regarding resident assistants, resident assistant directors, and professional staff's phone numbers, room number, and email address. The reason being for this is so that the staff can have a mobile and easy way to access personal information for an attempting to contact another resident assistant, resident assistant director, and or professional staff (for emergency purposes). This information being public has been a major security issue for quite some time, and we wanted to dedicate our project to making sure that our directory was secured. Furthermore, we wanted to make sure that the importance of this auxiliary web application was to protect the information of Residence Life staff. The site would be accessed through Okta, which would require a valid password and username that would be strictly allowed access to Professional Staff, Resident Assistant Directors, and Resident Assistants. By taking this very important precautionary step, this will limit the access of professional staff information to strictly Residence Life staff rather than being open to the public.

Stakeholder Input & Technical Requirements

Shortly after deciding on our project plan, our group worked hard to meet with essential stakeholders to be able to get the resources that we would need to make this project possible. Ryan Roper and Maxwell Kizewski were tasked with meeting with the essential personnel in this

project as they are both Resident Assistant Directors and already have a positive communication with Professional Staff. On January 24th, they met with Executive Director of Campus Life Sean VanGuilder, Assistant Director of Residence Life for Student Conduct Sean Ferguson, and Associate Vice President of Student Affairs Ken Posner to get approval for the project. Sean VanGuilder would later end up being our main point of contact for resources and connections as we worked forward on completing our project. In this stakeholder meeting, we discussed in depth our plans for the project and found that there was a great deal of excitement and skepticism on a student made application. These can all be seen in Appendix A as well.

After this meeting, our group held a meeting during our class time together to discuss the foreseeable limitations and problems that could prevent us from completing our project. A major concern that we received from our stakeholders was overall maintenance of the site. This concern came from AVP Ken Posner, and stated that students had made an application in a capstone class when he worked at Rollins College, and the application was excellent and useful but poorly maintained after the students had left. Therefore, he wanted us to make sure that the question of how future administration and maintenance of the web application would occur. He wanted to make sure that we kept this in mind as we built our application, and this would later come as a major factor when pitching our website to TI3. This was why we wanted to make sure our website was easily understandable and able to be managed by non-technically inclined individuals. This task was imperative in making sure that we would have a complete product that would be accepted by the Office of Residence Life and Student Affairs.

In our meetings, we also got a few concerns about how to make sure we get all the resources we need to allow our website to operate. We would need a server to be able to test out

Okta account making and make sure that our website would be created in a transferable manner. Jared DeWinne suggested that the best course of action would be to utilize Amazon Web Services as this would offer a cloud solution for our server needs. To ensure this would meet our needs, this required us to utilize the services of Amazon Route 53 and GoDaddy to purchase a domain name, and static IP for our AWS instance. We also initially had the idea of making a strictly mobile application that would be used by our Resident Assistants, which would involve us to use Swift and Objective-C. However, due to the overall nature of the application and what we wanted to do, we found that a web application would be a much better option. We also were able to be provided a server by our instructor Dr. Van Nguyen, which allowed us to host the website and Jared DeWinne used a spare domain that he already had. Therefore, many of our preliminary plans of action were scrapped after this meeting and the group decided to go in a different direction.

After consulting with Dr. Van Nguyen, the group decided to take a more focused approach of centralizing our application using a local server-based alternative called Node.js. Node.js proved to be immensely powerful tool that allowed us to run a localized environment for all of our JavaScript, CSS and HTML code to interact with each other. This solution consolidated a lot of our issues that we were originally going to face, and allowed us to create a demonstrative application that would be shown to our stakeholders and TI3 later on in our project development.

Project Planning

As there are a lot of different moving parts in this project, our group knew that the distribution of roles would be essential in the successful completion of the web application. Therefore, we decided to focus on individual strengths of each member of the group. Jared

DeWinne was assigned to focus on the back end development of the web application. This included securing and configuring a server to host testing operations, making connections with Okta to allow for account testing, creating a swap feature for duty swapping, and creating a calendar that would be able to be used by the Resident Assistants to know when they were scheduled for duty. Maxwell Kizewski was put in charge of the front end of the web application by creating most of the user interface that the Resident Assistants would access. Maxwell was also tasked with creating the duty check-in feature, the on-call quick access interface, and an easy directory access with all the personal contact information that would be needed by Residence Life staff. Ryan Roper was put in charge of documentation, in which all notes and papers that would be required for the class would be directed by him. He was also in charge of communicating with stakeholders. Notice that these roles fluctuated throughout the entire development process.

As a part of the planning process, a great deal of research and preparation was done to make sure we would create obtainable goals. We decided that due to the short nature of the class, we would utilize the waterfall model that was detailed in our opening presentation of this class. This would allow us to stay on a linear trajectory and plan for the project accordingly. The original project plan was completed by Maxwell Kizewski utilizing an interactive Gantt chart (See Appendix B). Ryan Roper was tasked with creating the wireframes for the entire website and incorporating all of the different roles of the application's interface (See Appendix C). Jared DeWinne was then immediately tasked with doing research on how to create a calendar application as well as how to set up the server with an Okta developer account to make sure we knew how to direct TI3 in setting up the new web application.

Node.js Front End Development

For the capstone project, Maxwell Kizewski's main goal was to create the user interface that would be appreciated and accepted by the Residence Life community at Saint Leo University. To do this, we were initially guided to utilize Node.js to help carry out most of our development. As nobody in the group had ever used Node.js before, the beginning of the project required tedious research in how to set up the Node.js environment. It took about a week to be able to build out the server environment through our website.js file (See Appendix D), but after following the tutorials presented to us by Dr. Nguyen, we were able to get the ecosystem working for localized testing. This is where the HTML development stage of the project began.

Maxwell utilized Adobe Dreamweaver as the Integrated Development Environment of choice as it can simultaneously support JavaScript, CSS, and HTML files while showing changes in a real time environment. This means that if a change was made in the code, it would immediately reflect those changes in the visualization provided by Dreamweaver. Let it be known that it was found early on that making edits through the visualization screen caused complications with the code, so we avoided directly editing elements and instead made changes with the code.

After setting up the coding interface, Maxwell downloaded a version of Bootstrap that was modestly built with custom containers that would make formatting the page easier. The Bootstrap was a part of a template called Visualize, which is released under a Creative Commons Attribution 3.0 Unported license, which allows for adaptations and transformations with reasonable recognition (Templated, 2016). In using this Bootstrap file, the page was able to encapsulate easy to manage containers for the homepage and supported custom images. Maxwell

arranged created and modified these containers into in a 3x2 matrix. He then included images were personally taken and edited through Adobe Photoshop. By using images as a branding piece, it was pretty important try to associate the look and feel of the images with the tasks that were being done (i.e. Directory Access is associated with the statue of Rudolph Antorch). It also makes the site interface more comfortable to use on both a PC and mobile environment.

To help ensure the branding of the page, Maxwell spent countless hours building out the user interface to meet the stakeholder's requests and map out the formatting in the bootstrap file. By taking these precautionary steps, we were able to create the color palette so it would reflect all the different pages that we would build out. This proved to be exceptionally useful work in the beginning as the changes made in the bootstrap file reflected globally and made the entire website look complete. Due to this, Maxwell made a conscious effort to steer away from using the <style> subsection in the HTML files as it was desired that the formatting of the site be restricted to the bootstrap file.

As for the HTML, the header and footer were all implemented and shared throughout the sites to make sure that the branding of the site would remain the same. The HTML code from the homepage of the website can be found in Appendix E to showcase the overall modeling of the page. After these preliminary elements of the webpage were successfully created, the focus of the project turned towards the duty check-in, directory access, and on-call quick access features as these were decided early on to be critical elements of the user experience.

The first page environment that was created was the duty check-in pages. These pages were basically a form that was dependent on a specific variable being selected—the location of where the RA was checking in for duty from. This made things fairly easy as when the site was

integrated with TI3's ecosystem, the group would be able to link user login credentials and attributes to the site, so the name would be automatically inherited. This layout arguably took the longest to build out as there was a need to set up a google email account enabled for third party access so an email could send from the site properly. This was done using Nodemailer, which is a built in mailing protocol created for Node.js and was organized through the website.js file that can be found in the Appendix. Likewise, a page map of how to check in for duty can be seen in Appendix F. In order to do this properly, two excellent tutorials were used to help guide the creation of this email system (Accime, 2019 & Traversy Media, 2017). These tutorials ultimately lead to the success of the webpages. To demonstrate the success of these pages, in Appendix G there is an example of this implementation with duty check in for Apartments 1+2.

The next page of interest was the Directory Access. As this has been a concern for TI3 for quite some time, this part of the web application was taken heavily into account. The Directory Access is essentially just a table view of all Residence Life staff member personal information. The original intention of this was to tie it with user accounts, but as user accounts would be changed regularly and displaying them in a table could be problematic as a security concern, we decided as a group to avoid doing this. To meet the standards of our stakeholders, this information is locked behind the login of the site, so that it is not able to be accessed out outside individuals unless they are accepted users through Okta's login. This will help preserve the contact information of our employees, as well as make contacting interpersonal contact outside of the workplace environment easier (See Appendix H).

The final page that Maxwell was directly responsible for was the On-Call Quick Access, which has unfortunately had been phased out of the final environment. There are times where

RAs will need to call Professional Staff or Resident Assistant Directors quickly, and calling the correct supervisor is always important as there are weekly shift rotations that each supervisor is responsible for. The purpose of this part of the application was to be tied into the calendar interface that was being built by Jared. Unfortunately, we decided that due to time restrictions that we would not further develop this part of the web application, even though a user interface has been built out for this part of the application (See Appendix I).

Node.js Back End Development

When Jared originally started the duty swap application, Jared attempted to look for something that already existed. What Jared ended up finding was a schedule swap application for employees in various organizations. Jared attempted to take on both the front end and back end of this application, making various adjustments along the way that suited the needs of Saint Leo Residence Life.

The aim of the swap project was to create an app that will allow Residence Life Pro Staff at Saint Leo to create and manage their RAs & RADs through shifts. It will also allow Residence Life workers to successfully and quickly swap shifts with their counterparts, without having to go through a third party, updating the schedule in the calendar. At the end of the adjustments, the following functionality would be available:

- Users can only see and request swaps with employees with the same job title as them
- Users can click a single button to request a shift swap with another employee's shift
- Users can view their notifications to see requests (shift swaps) from other users
- Users can approve or decline a shift swap via their notifications with a single click

The frontend application was tested and debugged with *Cypress* (Baxter, n.d.). This application was written for React, and combined the use of the backend, *Cypress*, and a calendar application to properly work (Jqense, 2020). The backend (ChChloeverity, n.d.) is a Ruby on Rails API, which communicates with the frontend using React to send requests to the API. The original authors also put an emphasis on utilizing React and JavaScript to improve the overall user experience.

The issues started to occur when Jared began to take the application out of development. As mentioned earlier, the code that was published on GitHub was implemented with “Cypress”, a JavaScript End to End Testing Framework. Jared tried to put all the React, Ruby on Rails API, and JavaScript into a Node.js Application that would easily host everything and be deployable wherever needed. Even the attempt to combine the different repositories took hours, and after an enormous amount of time it was decided that the time put in was not worth the end goal. The application was buggy in development mode, and never saw the light of day to make it for live testing.

Stakeholder Analysis & Wordpress Migration

After a few weeks after spring break, our group was able to coordinate a meeting with Information Security Engineer Greg Hodges, Developer/Integration Specialist Jason Gromley, and Executive Director of Campus Life Sean VanGuilder (the meetings from this meeting can be found in Appendix J). The goal of this meeting was to demonstrate the current state of our project as it was almost complete and discuss on boarding the application to be housed by TI3. After showing our application, TI3’s representatives were very impressed with the user interface and development, but they had demonstrated concerns about migration. TI3 has phased out

Linux based servers in their ecosystem since 2019, and as Node.js is specifically created to allow efficient hosting of the site on a Linux based device, it proved to be very difficult to convert the web application to the Windows environment. Also, maintenance of the site is difficult as the Node.js website will have to be manually updated and managed over time by TI3, which will not be easy as even the smallest change would have to be done through them, as Ken Posner had warned us about. After our first meeting with TI3, as a group we decided for those reasons to remediate this issue by converting our website over to a WordPress based system, which would ultimately make an easier transition into Okta and allow for non-technical individuals to edit the page themselves and ease the maintenance burden off of TI3's employees.

In creating the UI for WordPress, a great deal of the code we had already created was plug and play with a few modifications. We needed to remove the old header and footer that we had and replace it with one WordPress uses, which Jared was able to set up nicely. We also needed to modify the user environment so it would be adaptable and easily manageable long term by both TI3 and the Office of Residence Life. This was why we decided to use Elementor, which is a plugin for WordPress that allows for easy and simple editing of webpage interfaces. The use of Elementor has made things easy for integrating our current setup without too much of a loss to the "feel" of our Node.js pages, but solved our problem of having easy to manage pages long term.

Another backend process began. A Windows server was installed IIS, then WordPress was set as the default website. After getting the correct port forwarded, a rewrite of a record occurred for Jared DeWinne, which was a group member's spare domain, to the IP of the server owned by Professor Nguyen. Then it included port 5096 in the localhost binding set as that was

the port Professor Nguyen forwarded for HTTP access for the group. Once that was completed, the SAML plugin was implemented for having SAML2.0 authentication through Okta (See Appendix L & M). The site was locked down by unwanted and unknown users, having the default redirection site be the Okta page. By doing so, once you were redirected to the Okta Home Site, you then used the Single-Sign-On feature within Okta to seamlessly login to the RAAA application/website. A few users were created, adjusted, and tested in the Okta developer administration console. Different access levels were granted, given few users only reading and commenting permissions, and designated others Administrative Permissions. This was tested to completion by identifying if logged in users could then have an account made behind the scenes within the WordPress site. As shown in the “Welcome” page, once a user is logged on and goes to the RAAA application, they are greeted with “Welcome, %First Name%” where the First name is filled in with their’s, respectively. Default users within the WordPress Application can only read, and comment under designated posts. Future Admins of the site have to be added after the initial creation of their account within Okta and WordPress, thus adjusting their access permissions up to or equal to “root” (See Appendix N for Wordpress Site Code).

Results

After we completed our migration, our group ended up creating a website that has a dramatically more appealing look with new features that were not on the original resident assistant resource site. The new working features included a duty check-in function, a secure portal, and a monitored duty exchange platform. A live calendar function did not make the edition to the final site, so the Office of Residence Life has agreed to keep the current TeamUp calendar option for the time being. TI3 and Students Affairs was also very excited and

appreciative towards the efforts made creating the site. Both parties were so pleased in our work that they are hoping to use the web application our group created as the replacement for the resident assistant resource site. Meaning that the currently “rogue” resource site would be eliminated and that the auxiliary web application that was created in this class will now be the site where resident assistants will go to for online resources.

To elaborate on the decisions that were made to do this, Maxwell Kizewski attended a meeting scheduled by Sean VanGuilder to speak with administrative members at TI3. These members were ultimately concerned about two key factors: migration and long-term maintenance. Maxwell spoke about the stages of the website and how that it was originally created in a Node.js environment, but after our first meeting with TI3, the website took an alternate shift towards the Wordpress environment to be able to support service through a Windows IIS. As TI3 strictly uses an all Windows environment for servers, this was a very attractive feature of our final product. It was later explained to them that the website was practically plug and play, with only needing a domain, server, and active port to be configured. This meant that the migration process would be very easy for TI3, and they found this to be another excellent feature. TI3 agreed to house the website in-house without a problem, and they felt that this website would be an excellent setup for migrating the entire existing resource site to the new one.

As the site currently stands on the Wordpress environment, this proved to be a tad problematic from the maintenance side of administrating the site. TI3 utilizes HubSpot for all TI3’s major web application needs, which is a paid service that the university provides. Although TI3 would be able to accept and export the current site as is, they were concerned that in the next

5 years they would not have a good grasp of how to oversee the site and make sure that it is well maintained. Therefore, they requested that a migration would occur to transition the site from Wordpress to HubSpot. Maxwell stated that he was attempting to obtain a Graduate Assistant position at the university for the next few years and he agreed to take on the responsibility of helping transition these pages over to HubSpot and be TI3's point of contact for the Office of Residence Life.

At the end of the meeting, Maxwell had requested from CTO Manny Rodriguez and Sean VanGuilder that a letter of intent be written on behalf of TI3 and Student Affairs that this website would be applied, migrated and used by the Office of Residence Life in the coming months. They graciously agreed, and sent a letter of intent to Dr. Van Nguyen on April 16th, 2020 (See Appendix K). Overall, our group had very successful results and we created a product that will be implemented by the Office of Residence Life.

Discussion & Reflections

After reflecting on the finished product, we are extremely proud from how far we have come with the duration of the project. Goals that were made priority of the project have come into reality with hard work and using outside resources. As mentioned within the paper provided by TI3 and Student Affairs, they are eager to move forwarded with the website and utilize it as the main online resource for resident assistants. This accomplishment was not originally planned on when brainstorming or considered when originally taking on the web application, however it was later implemented and proves to be a testament to our dedication to creating a representable product for the University.

Another factor played into the creation and affected the timeline of the project was the Coronavirus. Coronavirus negativity impacted the timeline of the project by making it difficult for group members to meet due to complications moving out the residence halls and traveling home. Reflecting on the project even further, another untaken opportunity that would have made our project development easier would have been setting up a meeting with TI3 before developing the website. As TI3 was not interested in onboarding a linux-based system into their environment, the migration of our website into a Wordpress site was a decision that ultimately paid off. However, due to the timing of receiving that information late into the semester, we had to make tough decisions about the site that would ultimately change some of our features that we wanted to implement. Attempting a little more ambitiously to meet with TI3 in the beginning of the semester would have limited the different times of migrating the code and back end development to different environments. However, even though our group ran into this situation, we were able to create a product that was accepted by both parties we were aiming to impress.

Summary & Conclusions

To conclude, this web application is now up and functioning without any problems. New features have been added to give the users and administration jobs more accessible and more secure access. Although we had some ambitious plans for features that were to be created, due to different types of adversities the features were not developed fully. After coming back from spring break, the class was stopped for a week due to the Coronavirus. This affected team members being able to effectively meet up due to travel and limited technology access. Also, meetings with both the Student Affairs Office and TI3 became difficult due to the Coronavirus and campus being shut down. Despite the adversities, our web application with features that the

Office of Residence Life and TI3 wanted, is now approved and moving forward to install within the months to come.

This project gave members a firsthand experience of software development and understanding the teamwork and the techniques comprised of the topic. Many of us have never created an application from scratch before, so this entire process of working with different environments such as Okta, AWS, Node.js, Schej, Cypress, and Wordpress gave us a wide breath of different tools that helped us create a very useful web application. Our group also worked diligently to plan accordingly and work as close to the waterfall method as possible when creating our site. Based on our first class, we were made aware that commitment to a timeline would ultimately make us successful in completing our Capstone project. This class also gave the group members the opportunity to experience collaborating with professional organizations, such as TI3, the Office of Residence Life and the Office of Students Affairs at Saint Leo University. Indeed, we had several drawbacks that occurred throughout our project that ultimately caused different problems, but through these adverse conditions we were able to find success in the product we organically made as a team. Overall, the project was a glistening success, so much so that our finished product will now be transitioned into the main resource site for the Office of Residence Life to use moving forward.

Acknowledgements

Special consideration would like to be made to the many essential personnel that helped make this project possible. Dr. Manh Van Nguyen was the overseeing professor of the Computer Science Capstone course (COM-497) at Saint Leo University for Spring 2020. Dr. Nguyen provided the group with direct guidance as to how to best approach this project and ensured that

the group was supplied with adequate resources. Executive Director of Campus Life Sean VanGuilder provided faculty assistance and ensured that our project creation had the support of Student Affairs throughout the development process. Sean provided us with a great deal of feedback, kept in constant communication with us throughout our development stages and organized two meetings between the group and TI3. A great deal of appreciation goes to the individuals working in TI3, specifically CTO Manny Rodriguez, Information Security Engineer Greg Hodges, Director of Applications Daren McNab, Director of Information Security Darius Lewis, Director of Infrastructure Darren Cooper, and Developer/Integration Specialist Jason Gromley. All of these individuals took time out of their busy schedules to meet with us and offer their resources to make this application possible. Without their assistance and willingness to work with our team, this application's deployment for actual use by Saint Leo University's Office of Residence Life staff would not be possible.

Appendix A - Proposal

Residence Life Web Application Proposal

Submitted January 20th, 2020

After a discussion within the group, we have decided to create a web application for Saint Leo University's Office of Residence Life. Two members within our group work within the department and feel as though there could be a great need for a website where there will be additional resources for Resident Assistants and Residence Life team. As a Resident Assistant, aspects of the job should be easily accessible. There are moments where seconds count in the position; which means centralizing and integrating new tools for RAs to better do their jobs in a quick timeframe would be extremely helpful. Therefore, our group is planning to create an auxiliary website to lighten the load of the average Resident Assistant and the rest of the Residence Life staff.

Our web application will bring together all the necessary components of the Resident Assistant position into one, centralized location. Currently the position has several interfaces that are used, which results in the use of several different links that are sporadically located. Our goal is to centralize these tools that the RAs currently use and introduce new applications into their user ecosystem to make their jobs easier. This application would result in being a central location for: Current RA Website, Duty Log Calendar, Program Calendar, JAMS/Simplicity. New features that could potentially be added include: Instant Duty Swap, Duty Check-in, On-call Quick Access Phone Links.

In order to make this possible, our group will be working closely with Assistant Director of Residence Life for Student Conduct Sean Ferguson, Executive Director of Campus Life Sean VanGilder, and Associate Vice President of Student Affairs Ken Posner to get a better idea of what they would like to see from an advisory perspective. We will also host focus groups with Residence Life staff to help define boundaries for our project. Overall, we'll use the waterfall software development cycle to keep us organized and on track during this process.

(i) STAKEHOLDER REQUIREMENTS

Max and Ryan held a Stakeholder meeting on Friday, January 24th. The meeting consisted of mainly supervisors and professional staff for the Office of Residence Life, in which we surveyed their desires for what they would like to see in a new application. It seemed like there was a lot of positive feelings towards the idea of a new application that were in conjunction with the hope that key elements of the current infrastructure are not changed too much.

The consensus seems to be that there is a need for some type of application that simplifies and mainstreams RA accessibility among all its platforms. Several applications are used simultaneously in the position, so implementing an application where these can be used together would be beneficial and efficient. Below are the do's and don'ts from the meeting that we had with our stakeholders, as well as new features that would be desired in the application.

- Do's and Don'ts
 - Don't change too much of the existing infrastructure of the websites
 - Easy to maintain

- User friendly
- Easy accessibility online (tied to OKTA)
- Easy Mobile User Interface and Desktop Interface for RAs & Professional Staff
- Admin. Access for Professional Staff
- Key Features
 - Access to currently used applications: (JAMS, Residence, RA Website, TeamUp (Duty), TeamUp (Programming))
 - Duty Swapping System
 - Duty Check-in System (via Text or Email)
 - Lockout Database [NEW UPON STAKEHOLDER MEETING]
 - On-Call Quick Access Phone Links
 - RA/RAD/Prostaff Mobile Directory Access (Email & Phone #)

It should be noted that this is a list of desired features that would like to be seen by the staff.

Given time restraints, we may not be able to completely implement all of these features. This list of features should give us enough work to build a solid infrastructure around our project that is adequate for capstone and meets the needs of the Office of Residence Life at Saint Leo University.

(ii) LIST TECHINCAL REQUIREMENTS:

Based on our desire for a web-based application, that allows for speedy communication, and unmatched organization, we need a few services, and sections of knowledge to make this project possible. The table below represents our requirements, and the summary below attempts to explain each and every one of them.

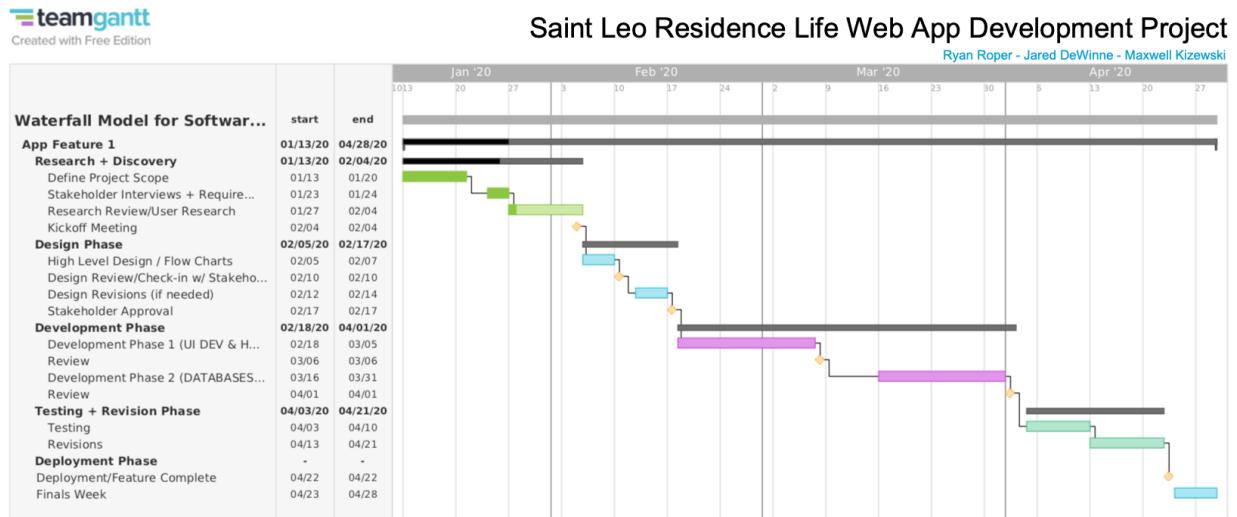
<u>Programming Languages</u>	<u>Online Services</u>	<u>Assistance/ Authorization from other entities</u>
Objective-C	AWS EC2 service	UTS permission to add program to Okta
Swift	Domain Hosting & DNS supplier (GoDaddy, Amazon Route 53)	IT permissions to setup access management for the Okta/Microsoft portal to authorize users
HTML	Microsoft Login Portal management	
CSS		
PHP		

Our project will feature a web-based application that authorized users can access from any platform they wish, either by downloading an app or going to our website. To make this possible, we need the above-listed services/requirements that will enable the finished project to work seamlessly with real data. First, we need to host the service. This will require server space/cloud equipment from a company such as AWS, which we are currently planning to use. After this is hosted, we need to ensure a constant DNS connection, by utilizing the services of Amazon Route 53 and GoDaddy to purchase a domain name, and static IP for our AWS instance. To meet

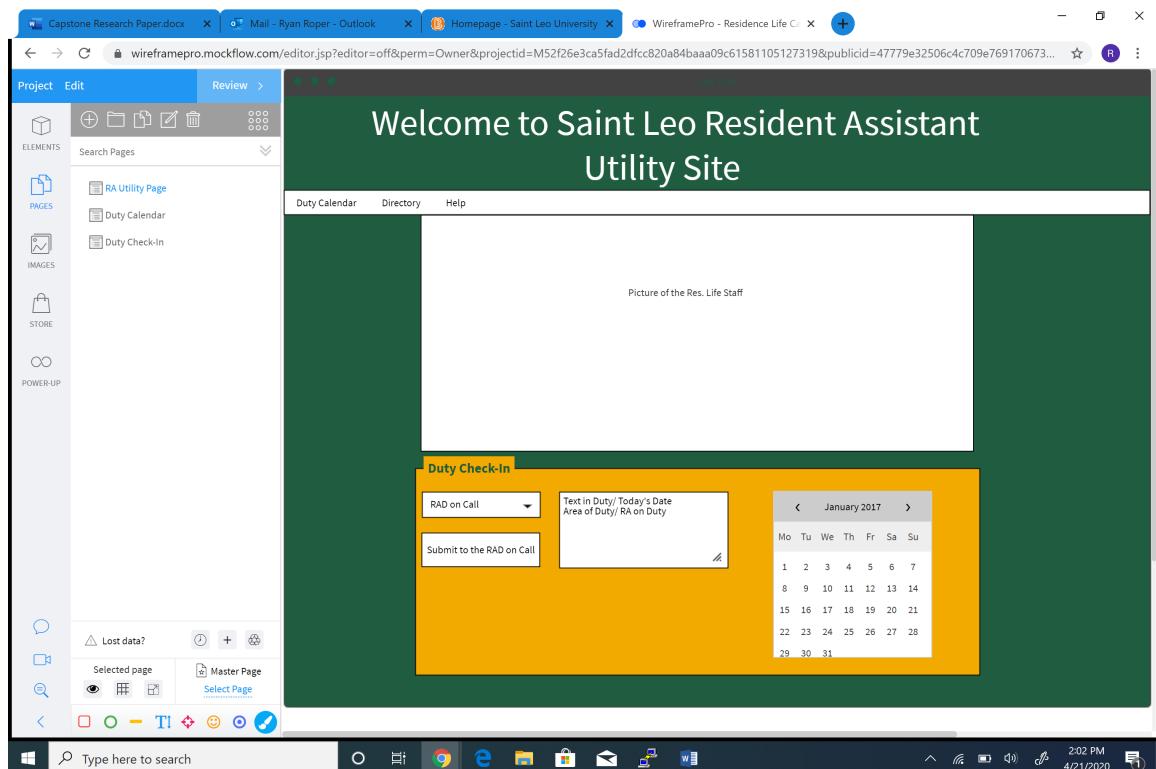
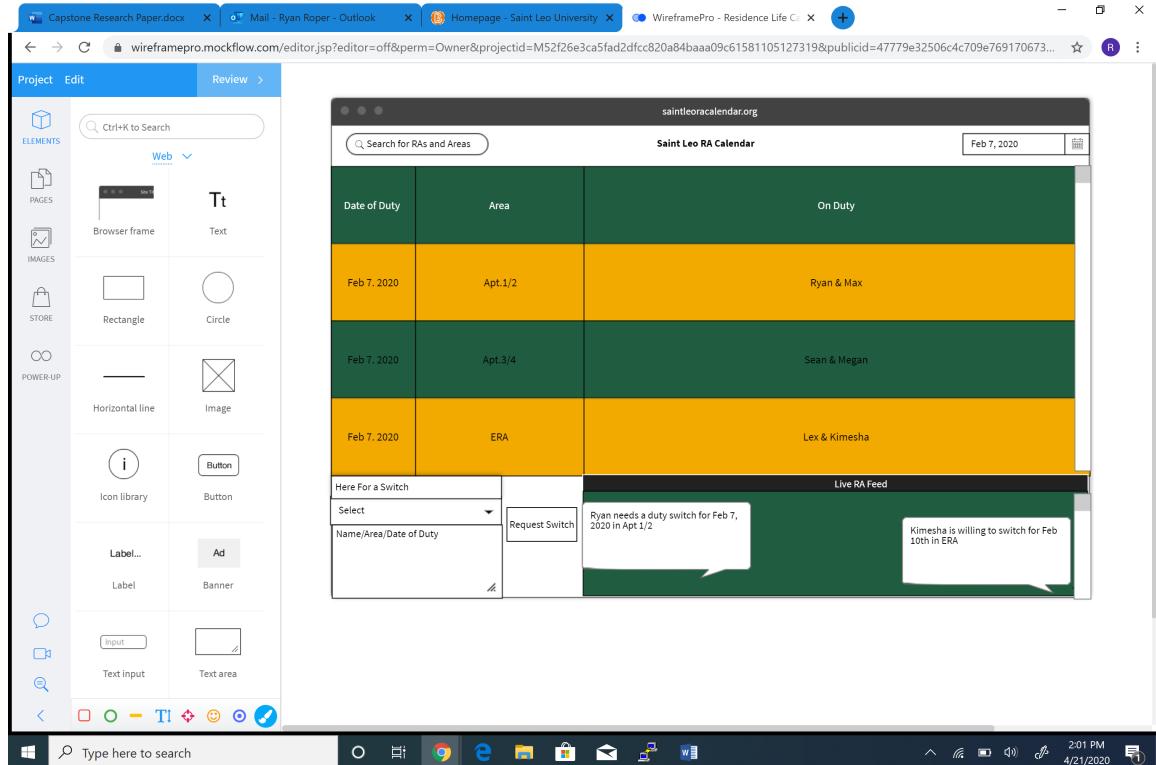
the goal for scalability, and customization we would need to enable on-demand scaling from AWS for our specific instance, allowing any means of traffic and data to be allocated for, without overspending for a server bigger than we need. The overall program we are planning on building is predicted to only use HTML and CSS, along with PHP, for any email functionality. The application by nature is simple, but how we are implementing it requires full accessibility from all entities. Using knowledge with Swift and Objective-C, we will create iOS and Android applications that connect to the mobile version of the program, such as using a Safari App Extension for iOS. Where user management comes into play, is where we need authorization from UTS/TI³ to change Okta and identify Microsoft User management. We want specific users (RA's/RAD's/etc.) to access the primary form of the platform, allowing them to use the features granted to them, while other users (their bosses/department heads) have management features where they can adjust functions such as user role management, and access permissions. This will require implementation with both Okta and Microsoft Portal since they are the two forms of user authorization used for Saint Leo's platforms.

Appendix B - Gantt Chart

Below you can see the Gantt chart that was created, emulating the Waterfall model for our project to ensure that we remained on time with completing tasks, meeting deadlines and interacting with stakeholders:



Appendix C - Wireframes



Appendix D - website.js

```
var express = require('express'),  
path = require('path'),  
nodemailer = require('nodemailer');  
app = express();  
  
// Nodemailer Data Transfer ----->  
app.use(express.urlencoded({extended:false}));  
app.use(express.json());  
  
app.post('/location', (req, res) => {  
    console.log('INCOMING DATA: ', req.body) // Console Log for terminal view  
    const output = `<h4>A MESSAGE FROM THE</h4><h2>Resident Assistant  
Accessibility Application</h2>  
    <p>A Resident Assistant has checked in for duty at ${req.body.location}:</p>  
    <h3>Resident Assistant: <i>userName</i></h3>  
    <h3>Location: <i>${req.body.location}</i></h3>  
    <h3>Time: <i>dateTimeEST</i></h3>  
    <p>For further information or questions, please contact <i>userName</i> directly.</p>  
    <br><p>Have a wonderful evening!</p>`;  
  
    // Transporter for Sending Nodemailer Data (U&P)  
  
    var transporter = nodemailer.createTransport({  
        service: 'gmail',  
        auth: {  
            user: 'saintleoresnotify@gmail.com',  
            pass: 'ThisIsATestPassw0rd!' // MUST BE HIDDEN!  
    }  
});
```

```
        }  
    });  
  
// Nodemailer Mail Sending Setup ----->  
  
var mailOptions = {  
  
    from: '"Resident Assistant Accessibility Application" <saintleoresnotify@gmail.com>',  
    to: 'maxwell.kizewski@email.saintleo.edu',  
    subject: `Duty Check-In for ${req.body.location}`,  
    html: output  
};  
  
transporter.sendMail(mailOptions, function(error, info){  
    if (error){  
        console.log('ERROR: UNABLE TO COMMUNICATE WITH SERVER |  
smtp.gmail.com');  
    } else {  
        console.log('EMAIL SENT: ' + info.response);  
    }  
});  
});  
  
// Express Middleware ----- Static Files ---->  
  
app.use(express.static(path.join(__dirname, 'public')));  
app.get('/', function(req, res) {  
    res.redirect('index.html');  
});  
app.listen(3000); // Listen Port 3000
```

Appendix E - index.html (Home Page)

The screenshot shows the homepage of the Resident Assistant Accessibility Application (RAAA) for Saint Leo University. At the top, there's a dark header bar with the university's logo and navigation links for 'Home', 'Duty Check-In', and 'Quick Links'. Below the header, a large welcome message reads: 'Welcome, *userName*', followed by 'The Resident Assistant Accessability Application for Saint Leo University'. A subtitle below states 'Enabling Resident Assistants with Easily Accessible Tools Essential to Succeed in Their Role'. The main content area features two prominent sections: 'Duty Log Calendar' (with an image of a bronze lion statue) and 'Duty Shift Exchange' (with an image of a building). Each section has a brief description and a 'View [Section]' button.

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>RAAA - Home</title>

    <!-- Bootstrap -->
    <link href="css/bootstrap-4.4.1.css" rel="stylesheet">

    <style>
      #brand-image
      {
        height: 30px;
      }

```

```
        }
```

```
    #brand-image2
```

```
    {
```

```
        height: 100px;
```

```
    }
```

```
    #wrap
```

```
    {
```

```
        background-color: #004900
```

```
    }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
```

```
    <a class="navbar-brand">
```

```
        
```

```
    </a>
```

```
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
```

```
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
```

```
expanded="false" aria-label="Toggle navigation">
```

```
        <span class="navbar-toggler-icon"></span>
```

```
    </button>
```

```
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
```

```
        <ul class="navbar-nav mr-auto">
```

```
            <li class="nav-item active">
```

```
                <a class="nav-link text-white" href="index.html">Home<span class="sr-
```

```
only">(current)</span></a>
```

```
            </li>
```

```
<li class="nav-item">
    <a class="nav-link text-white" href="DutyCheck.html">Duty Check-In</a>
</li>

<li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle text-white" href="#" id="navbarDropdown"
role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        Quick Links
    </a>
    <div class="dropdown-menu" aria-labelledby="navbarDropdown">
        <a class="dropdown-item" href="http://ras.saintleoresidencelife.com/">
            rel="noopener noreferrer" target="_blank">RA Website</a>
        <a class="dropdown-item" href="https://saintleo-advocate.symplicity.com/
index.php/" rel="noopener noreferrer" target="_blank">JAMS</a>
            <a class="dropdown-item" href="https://slu-
residence.symplicity.com/index.php/pid648336?#" rel="noopener noreferrer"
target="_blank">Residence</a>
            <a class="dropdown-item" href="https://teamup.com/
ks5d76f23008146a4d" rel="noopener noreferrer" target="_blank">Program Calendar</a>
        </div>
    </li>
</ul>
</div>

</nav>

<div class="jumbotron jumbotron-fluid text-center">
    <h1 class="display-4">Welcome, <b><i>userName</i></b></h1>
    <h2 class="display-4">The Resident Assistant Accessability Application for&nbsp;</h2>
        <h1 class="display-4">Saint Leo University</h1>
<div class="container">
    <hr>
```

<p class="lead">Enabling Resident Assistants with Easily Accessible Tools
Essential to Succeed in Their Role

<div class="row">

<div class="col-md-4 col-xl-6">

<div class="card">

<div class="card-body">

<h4 class="card-title text-center">Duty Log Calendar</h4>

<p class="card-text text-center">Access the Duty Log Calendar through this link to know who is on duty!

View Calendar

</div>

</div>

</div>

<div class="col-md-4 col-xl-6">

<div class="card">

<div class="card-body">

<h4 class="card-title text-center">Duty Shift Exchange

<p class="card-text text-center">Easily swap duties with other Resident Assistants to ensure you receive coverage!

View Exchange </div>

</div>

```
</div>
<div class="col-md-4 col-xl-4">
  <div class="card"> 
    <div class="card-body">
      <h4 class="card-title text-center">Directory Access </h4>
      <p class="card-text text-center">Access the contact information of all Resident
Assistants, Resident Assistant Directors and Professional Staff in this directory! </p>
      <br>
      <a href="DirectoryAccess.html" class="btn btn-primary">View Directory </a>
    </div>
  </div>
</div>
<div class="col-md-4 col-xl-4">
  <div class="card"> 
    <div class="card-body">
      <h4 class="card-title text-center">On-Call Quick Access</h4>
      <p class="card-text text-center">Need help with a situation? Quickly call the On-Call
Resident Assistant Director or Professional Staff using our Quick Access! </p>
      <br>
      <a href="OnCall.html" class="btn btn-primary">Contact On-Call Personnel </
a></div>
  </div>
</div>
<div class="col-md-4 col-xl-4">
  <div class="card"> 
    <div class="card-body">
```

```
<h4 class="card-title text-center">Duty Check-In</h4>
<p class="card-text text-center">Utilize the automatic check-in feature to sign in for
duty! Simply select your duty area and follow the instructions on the screen!&nbsp;</p>
<br>
<a href="DutyCheck.html" class="btn btn-primary">Check-In for Duty&nbsp;</a></
div>
</div>
</div>
<br><br>
<hr>
</div>
<br><br><br>
</div>
</div>

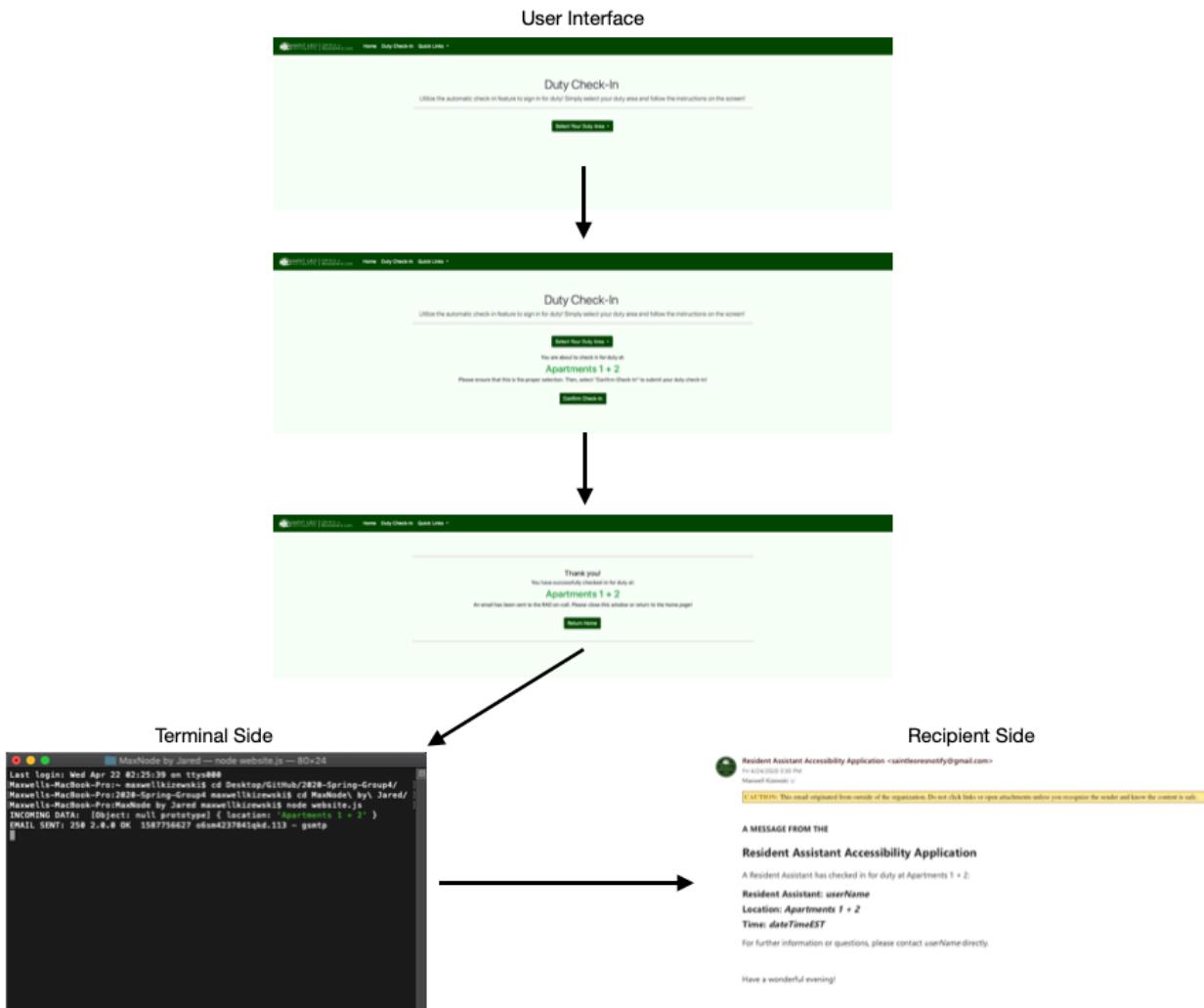
<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script src="js/jquery-3.4.1.min.js"></script>

<!-- Include all compiled plugins (below), or include individual files as needed -->
<script src="js/popper.min.js"></script>
<script src="js/bootstrap-4.4.1.js"></script>
</body>
<!-- Footer -->

<footer class="footer text-white" id="wrap">
<table align="center">
<col width= "100px">
<col width= "400px">
<tr>
```

```
<td></td>
<td valign="middle" class="text-left">Saint Leo University | Office of Residence
Life<br><i>33701 State Road 52 | Saint Leo, FL 33574<br>Main Office: <a href="tel:
3525888268" class="text-white">(352) 588-8268</a></i><br>"Love To Live Here"</td>
</tr>
</table>
</footer>
</html>
```

Appendix F - Duty Check-in User Effort Diagram



Appendix G - DutyCheck.html, DC_12.html, DC_12Confirm.html (Duty Check-in)

The screenshot shows a web page titled "Duty Check-In". At the top left is the Saint Leo University logo with the text "SAINT LEO UNIVERSITY | OFFICE of RESIDENCE LIFE". To its right are navigation links: "Home", "Duty Check-In", and "Quick Links". Below the header, the main title "Duty Check-In" is centered. A sub-instruction "Utilize the automatic check-in feature to sign in for duty! Simply select your duty area and follow the instructions on the screen!" is displayed. A prominent green button labeled "Select Your Duty Area" is centered below the instruction. At the bottom of the page is a dark footer bar containing the university's address: "Saint Leo University | Office of Residence Life", "33701 State Road 52 / Saint Leo, FL 33574", "Main Office: (352) 588-8268", and the slogan "Love To Live Here".

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>RAAA - Duty Check-In</title>

    <!-- Bootstrap -->
    <link href="css/bootstrap-4.4.1.css" rel="stylesheet">

    <style>
        #brand-image
```

```
{  
    height: 30px;  
}  
  
#menu1  
{  
    background-color: #004900;  
    font-color: #fff;  
}  
  
#brand-image2  
{  
    height: 100px;  
}  
  
#wrap  
{  
    background-color: #004900  
}  
  
</style>  
  
</head>  
  
<body>  
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">  
        <a class="navbar-brand">  
              
        </a>  
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-  
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-  
expanded="false" aria-label="Toggle navigation">  
            <span class="navbar-toggler-icon"></span>
```

```
</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">
  <ul class="navbar-nav mr-auto">
    <li class="nav-item active">
      <a class="nav-link" href="index.html">Home<span class="sr-only">(current)</span></a>
    </li>
    <li class="nav-item">
      <a class="nav-link text-white" href="DutyCheck.html">Duty Check-In</a>
    </li>
    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle text-white" href="#" id="navbarDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        Quick Links
      </a>
      <div class="dropdown-menu" aria-labelledby="navbarDropdown">
        <a class="dropdown-item" href="http://ras.saintleoresidencelife.com/" rel="noopener noreferrer" target="_blank">RA Website</a>
        <a class="dropdown-item" href="https://saintleo-advocate.symplicity.com/index.php" rel="noopener noreferrer" target="_blank">JAMS</a>
        <a class="dropdown-item" href="https://slu-residence.symplicity.com/index.php/pid648336?#" rel="noopener noreferrer" target="_blank">Residence</a>
        <a class="dropdown-item" href="https://teamup.com/ks5d76f23008146a4d" rel="noopener noreferrer" target="_blank">Program Calendar</a>
      <!----<div class="dropdown-divider"></div>---->
      </div>
    </li>
    <li class="nav-item">
```

```
<!---<a class="nav-link disabled" href="#">Disabled</a> --->
</li>
</ul>
</div>
</nav>

<div class="jumbotron jumbotron-fluid text-center">
    <h1 class="display-4"></h1>
    <h1 class="display-4">Duty Check-In&nbsp;</h1>
    <div class="container">
        <p class="lead">Utilize the automatic check-in feature to sign in for duty! Simply
select your duty area and follow the instructions on the screen!&nbsp;</p>
        <hr><br>
        <div class="dropdown">
            <button class="btn btn-secondary dropdown-toggle" type="button" id="menu1" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                Select Your Duty Area
            </button>
            <div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
                <a class="dropdown-item" href="DC_12.html">Apartments 1 + 2</a>
                <a class="dropdown-item" href="DC_34.html">Apartments 3 + 4</a>
                <a class="dropdown-item" href="DC_56.html">Apartments 5 + 6</a>
                <a class="dropdown-item" href="DC_RA.html">Roderick and Alumni</a>
                <a class="dropdown-item" href="DC_HB.html">Henderson and Benoit</a>
                <a class="dropdown-item" href="DC_MS.html">Marmion and Snyder</a>
                <a class="dropdown-item" href="DC_SA.html">Sweetwater Apartments</a>
            <div class="dropdown">
                </div>
            </div>
        </div>
```

```
<br><br><br><br><br><br><br><br>

</div>

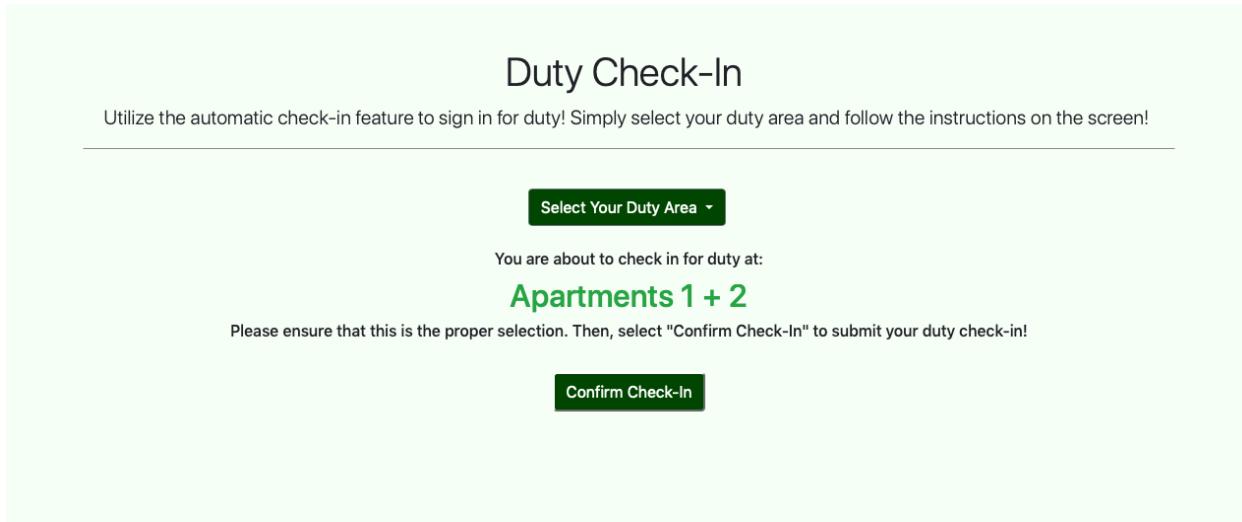
<br>

<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script src="js/jquery-3.4.1.min.js"></script>

<!-- Include all compiled plugins (below), or include individual files as needed -->
<script src="js/popper.min.js"></script>
<script src="js/bootstrap-4.4.1.js"></script>
</body>

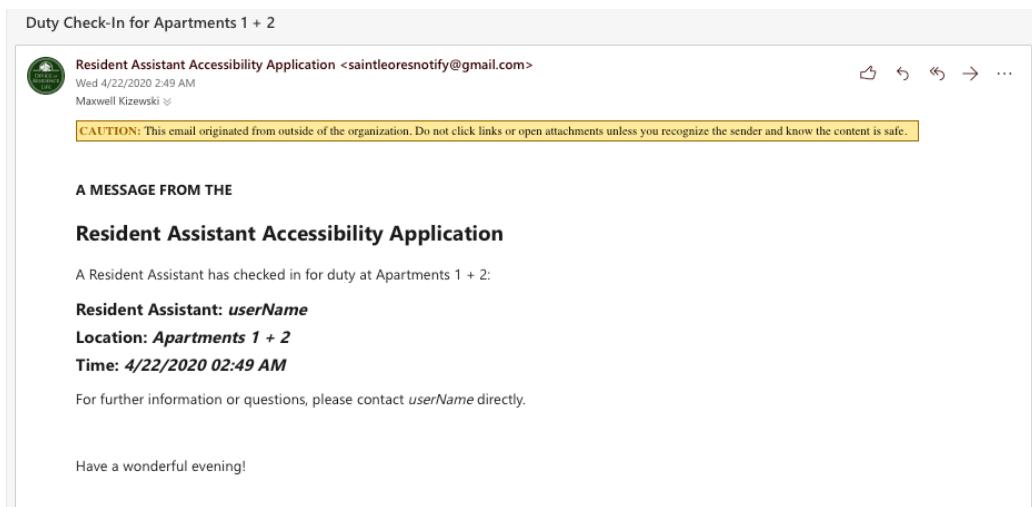
<!-- Footer -->

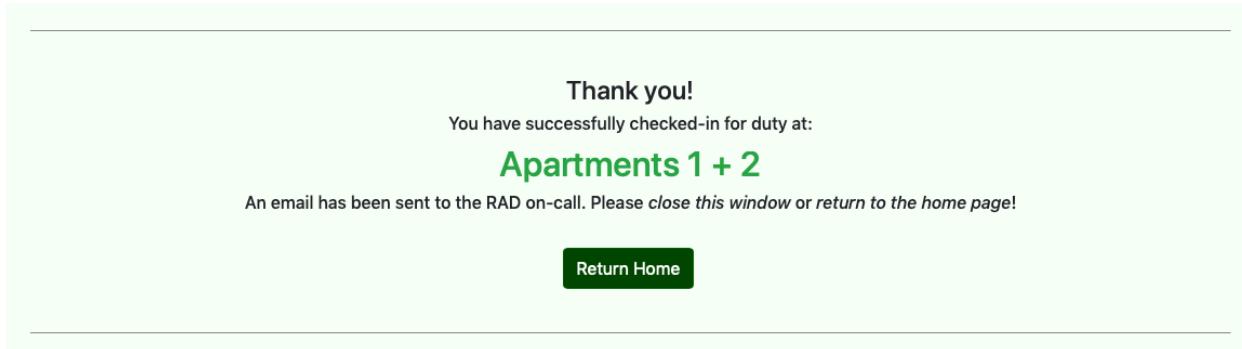
<footer class="footer text-white" id="wrap">
    <table align="center">
        <col width= "100px">
        <col width= "400px">
        <tr>
            <td></td>
            <td valign="middle" class="text-left">Saint Leo University | Office of Residence Life<br><i>33701 State Road 52 | Saint Leo, FL 33574<br>Main Office: <a href="tel:3525888268" class="text-white">(352) 588-8268</a></i><br>"Love To Live Here"</td>
        </tr>
    </table>
</footer>
</html>
<! - - - Begin Code Clipping - - - >
```



```
<button class="btn btn-secondary dropdown-toggle" type="button" id="menu1" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    Select Your Duty Area
</button>
<div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
    <a class="dropdown-item" href="DC_12.html">Apartments 1 + 2</a>
    <a class="dropdown-item" href="DC_34.html">Apartments 3 + 4</a>
    <a class="dropdown-item" href="DC_56.html">Apartments 5 + 6</a>
    <a class="dropdown-item" href="DC_RA.html">Roderick and Alumni</a>
    <a class="dropdown-item" href="DC_HB.html">Henderson and Benoit</a>
    <a class="dropdown-item" href="DC_MS.html">Marmion and Snyder</a>
    <a class="dropdown-item" href="DC_SA.html">Sweetwater Apartments</a>
<div class="dropdown">
</div></div><br><br>
<h6>You are about to check in for duty at:</h6>
<h2 class= "text-success">Apartments 1 + 2</h2>
<h6>Please ensure that this is the proper selection. Then, select "Confirm Check-
In" to submit your duty check-in!</h6>
<!-- FORM FORMATTING AT UI -->
```

```
<form>
    <input type="text" id="location" placeholder="Location"
value="Apartments 1 + 2" style="display: none" readonly> <br>
    <input type="submit" value="Confirm Check-In">
</form>
<!-- END FORM FORMATTING AT UI -->
<!-- SCRIPTS FOR EMAIL DEPLOYMENT -->
<script src="js/jquery-3.4.1.min.js"></script>
<script>
    $('form').on('submit', (e) => {
        e.preventDefault();
        const location = $('#location').val().trim(); const location = $(
            '#dateTimeEST').val().trim();
        const data = { location, dateTimeEST };
        $.post('/location', data, function() { });
        window.location.href = "DC_12Confirm.html"; // Redirect
    });
</script>
<!-- END SCRIPTS FOR EMAIL DEPLOYMENT -->
<! - - - END CODE CLIPPING - - - >
```





<! - - - BEGIN CODE CLIPPING - - - >

```
<div class="jumbotron jumbotron-fluid text-center">
    <!--<h1 class="display-4"></h1>
    <h1 class="display-4">Duty Check-In&nbsp;</h1>-->
    <div class="container">
        <!--<p class="lead">Utilize the automatic check-in feature to sign in for duty!
        Simply select your duty area and follow the instructions on the screen!&nbsp;</p>-->
    <hr><br>
        <h4>Thank you!</h4>
        <h6>You have successfully checked-in for duty at:</h6>
        <h2 class= "text-success">Apartments 1 + 2</h2>
        <h6>An email has been sent to the RAD on-call. Please <i>close this window</i>
        or <i>return to the home page</i>!</h6><br>
        <a href="index.html" class="btn btn-primary">Return Home</a>
        <!--<a href="#" onclick="close_window();return false;" class="btn btn-
        primary">Close Window</a>-->
    <br><br><hr><br><br><br><br>
</div>

<br>
```

<! - - - END CODE CLIPPING - - - >

Appendix H - DirectoryAccess.html (Directory Access)

 SAINT LEO UNIVERSITY | OFFICE OF RESIDENCE LIFE Home Duty Check-In Quick Links ▾

Directory Access

Access the contact information of all Resident Assistants, Resident Assistant Directors and Professional Staff in this directory!

Professional Staff

Member	Position	Email	Phone Number
Sean VanGuilder	Executive Director of Campus Life	sean.vanguilder@saintleo.edu	(###) ###-####
VACANT	Director of Residence Life	n/A	n/A
Sean Ferguson	Assistant Director of Residence Life: Student Conduct	sean.ferguson@saintleo.edu	(###) ###-####
Kimesha Norris	Assistant Director of Residence Life: Housing Operations	kimesha.norris@saintleo.edu	(###) ###-####
Megan Lawson	Graduate Assistant: Recruitment and Training	megan.lawson02@email.saintleo.edu	(###) ###-####
Lex Hauryluck	Graduate Assistant: Facilities	alexander.hauryluck@email.saintleo.edu	(###) ###-####
Kari Grippo	Graduate Assistant: Programming	kari.grippo@email.saintleo.edu	(###) ###-####
Jasmine Collins	Student Crisis Support Specialist	jasmine.collins@saintleo.edu	(###) ###-####

Apartments 1 + 2

Staff Member	Position	Email	Phone Number
Ryan Roper	Resident Assistant Director	ryan.roper@email.saintleo.edu	(###) ###-####
Staff Member #1	Resident Assistant	n/A	n/A

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>RAAA - Directory Access</title>
```

```
<!-- Bootstrap -->
<link href="css/bootstrap-4.4.1.css" rel="stylesheet">
<style>
    #brand-image { height: 30px; }
    #brand-image2 { height: 100px; }
```

```
#wrap { background-color: #004900; }
th, td { border-collapse: collapse; }
table#t01 th { background-color: #004900; color: white; }

</style>

</head>

<body>

    <style></style>

    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <a class="navbar-brand">
            
        </a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarSupportedContent">
            <ul class="navbar-nav mr-auto">
                <li class="nav-item active">
                    <a class="nav-link text-white" href="index.html">Home<span class="sr-
only">(current)</span></a>
                </li>
                <li class="nav-item">
                    <a class="nav-link text-white" href="DutyCheck.html">Duty Check-In</a>
                </li>
                <li class="nav-item dropdown">
                    <a class="nav-link dropdown-toggle text-white" href="#" id="navbarDropdown"
role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
```

Quick Links

<div class="dropdown-menu" aria-labelledby="navbarDropdown">
 <a class="dropdown-item" href="http://ras.saintleoresidencelife.com/"
 rel="noopener noreferrer" target="_blank">RA Website
 <a class="dropdown-item" href="https://saintleo-advocate.symplicity.com/
index.php/" rel="noopener noreferrer" target="_blank">JAMS
 <a class="dropdown-item" href="https://slu-
residence.symplicity.com/index.php/pid648336?#" rel="noopener noreferrer"
 target="_blank">Residence
 <a class="dropdown-item" href="https://teamup.com/
ks5d76f23008146a4d" rel="noopener noreferrer" target="_blank">Program Calendar
 <!----<div class="dropdown-divider"></div>---->
 </div>

 <li class="nav-item">
 <!---Disabled --->

</div>
</nav>

<div class="jumbotron jumbotron-fluid text-center">
 <h1 class="display-4"></h1>
 <h1 class="display-4">Directory Access</h1>
<p class="lead">Access the contact information of all Resident Assistants, Resident Assistant
Directors and Professional Staff in this directory!</p>
<div class="container">
 <hr>

```
<br>
<center><table style="width:100% id="t01" border="1px solid black
cellpadding="10px">
<h3><i>Professional Staff</i></h3>
<tr>
<th>Member</th>
<th>Position</th>
<th>Email</th>
<th>Phone Number</th>
</tr>
<tr>
<td>Sean VanGuilder</td>
<td>Executive Director of Campus Life</td>
<td>sean.vanguard@saaintleo.edu</td>
<td>(####) ####-#####</td>
</tr>
<tr>
<td>VACANT</td>
<td>Director of Residence Life</td>
<td>n/A</td>
<td>n/A</td>
</tr>
<tr>
<td>Sean Ferguson</td>
<td>Assistant Director of Residence Life: Student Conduct</td>
<td>sean.ferguson@saaintleo.edu</td>
<td>(####) ####-#####</td>
</tr>
<tr>
```

<td>Kimesha Norris</td>	
<td>Assistant Director of Residence Life: Housing Operations</td>	
<td>kimesha.norris@saintleo.edu</td>	
	<td>(###) ###-####</td>
</tr>	
<tr>	
<td>Megan Lawson</td>	
<td>Graduate Assistant: Recruitment and Training</td>	
<td>megan.lawson02@email.saintleo.edu</td>	
	<td>(###) ###-####</td>
</tr>	
<tr>	
<td>Lex Hauryluck</td>	
<td>Graduate Assistant: Facilities</td>	
<td>alexander.hauryluck@email.saintleo.edu</td>	
	<td>(###) ###-####</td>
</tr>	
<tr>	
<td>Kari Grippo</td>	
<td>Graduate Assistant: Programming</td>	
<td>kari.grippo@email.saintleo.edu</td>	
	<td>(###) ###-####</td>
</tr>	
<tr>	
<td>Jasmine Collins</td>	
<td>Student Crisis Support Specialist</td>	
<td>jasmine.collins@saintleo.edu</td>	
	<td>(###) ###-####</td>
</tr>	

```
</table></center>
<br><br>
<center><table style= width:100% id="t01" border= 1px solid black cellpadding="10px">
<h3><i>Apartments 1 + 2</i></h3>
<tr>
<th>Staff Member</th>
<th>Position</th>
<th>Email</th>
<th>Phone Number</th>
</tr>
<tr>
<td>Ryan Roper</td>
<td>Resident Assistant Director</td>
<td>ryan.roper@email.saintleo.edu</td>
<td>(###) ###-####</td>
</tr>
<tr>
<td>Staff Member #1</td>
<td>Resident Assistant</td>
<td>n/A</td>
<td>n/A</td>
</tr>
<tr>
<td>Staff Member #2</td>
<td>Resident Assistant</td>
<td>n/A</td>
<td>n/A</td>
</tr>
<tr>
```

```
<td>Staff Member #3</td>
<td>Resident Assistant</td>
<td>n/A</td>
    <td>n/A</td>
</tr>
<tr>
    <td>Staff Member #4</td>
    <td>Resident Assistant</td>
    <td>n/A</td>
        <td>n/A</td>
</tr>
<tr>
    <td>Staff Member #5</td>
    <td>Resident Assistant</td>
    <td>n/A</td>
        <td>n/A</td>
</tr>
<tr>
    <td>Staff Member #6</td>
    <td>Resident Assistant</td>
    <td>n/A</td>
        <td>n/A</td>
</tr>
</table></center>
<br><br>
<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script src="js/jquery-3.4.1.min.js"></script>

<!-- Include all compiled plugins (below), or include individual files as needed -->
```

```
<script src="js/popper.min.js"></script>
<script src="js/bootstrap-4.4.1.js"></script>
</body>
<!-- Footer -->
<footer class="footer text-white" id="wrap">
    <table align="center">
        <col width= "100px">
        <col width= "400px">
        <tr>
            <td></td>
            <td valign="middle" class="text-left">Saint Leo University | Office of Residence
Life<br><i>33701 State Road 52 | Saint Leo, FL 33574<br>Main Office: <a href="tel:
3525888268" class="text-white">(352) 588-8268</a></i><br>"Love To Live Here"</td>
        </tr>
    </table>
</footer>
</html>
```

Appendix I - OnCall.html (On Call Quick Access)

 [Home](#) [Duty Check-In](#) [Quick Links](#) ▾

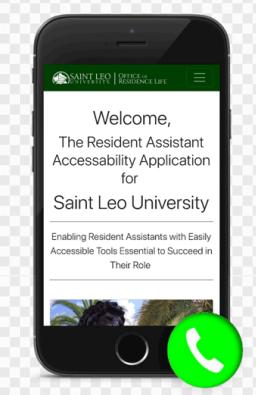
On-Call Quick Access

Need help with a situation? Quickly call the On-Call Resident Assistant Director or Professional Staff using our Quick Access!

When To Call Professional Staff

- Suicidal Gesturing
- Major Vandalism (Building door is broken, Flood, Fire, Causing harm or safety issues)
- Physical or Sexual Assault (including rumors of assault)
- Student Death or Injury
- Possession of Weapons
- Drugs/Paraphernalia
- Fire or Flood
- Student Transported to the Hospital

[Click to Contact Professional Staff On-Call](#)



When To Call RAD On-Call

- Pet Violations
- Vandalism (Graffiti, Small hole in a wall, Furniture damaged, Tagging)
- Small Roommate Disagreements
- Alcohol Found/Party
- Noise Complaints
- Cannot find Origin of Smell/Drugs Were NOT Found
- Toilet Won't Stop Flushing
- Visitation/Escort Policy (not endangering residents)

[Click to Contact Resident Assistant Director On-Call](#)

Please do not hesitate to call us if you need assistance, we are here to help you!

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>RAAA - Home</title>

    <!-- Bootstrap -->
    <link href="css/bootstrap-4.4.1.css" rel="stylesheet">

    <style>
      #brand-image
      {
        height: 30px;
      }
    </style>
  
```

```
        }
```

```
#brand-image2
```

```
{
```

```
    height: 100px;
```

```
}
```

```
#brand-image3
```

```
{
```

```
    height: 300px;
```

```
}
```

```
#wrap
```

```
{
```

```
    background-color: #004900;
```

```
}
```

```
#button-box
```

```
{
```

```
    text-align: center;
```

```
    margin-top: 10px;
```

```
    margin-bottom: 30px;
```

```
    background-color:
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
```

```
    <a class="navbar-brand">
```

```
        
```

```
    </a>
```

```
<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">  
    <span class="navbar-toggler-icon"></span>  
</button>  
  
<div class="collapse navbar-collapse" id="navbarSupportedContent">  
    <ul class="navbar-nav mr-auto">  
        <li class="nav-item active">  
            <a class="nav-link text-white" href="index.html">Home<span class="sr-only">(current)</span></a>  
        </li>  
        <li class="nav-item">  
            <a class="nav-link text-white" href="DutyCheck.html">Duty Check-In</a>  
        </li>  
        <li class="nav-item dropdown">  
            <a class="nav-link dropdown-toggle text-white" href="#" id="navbarDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">  
                Quick Links  
            </a>  
            <div class="dropdown-menu" aria-labelledby="navbarDropdown">  
                <a class="dropdown-item" href="http://ras.saintleoresidencelife.com/" rel="noopener noreferrer" target="_blank">RA Website</a>  
                <a class="dropdown-item" href="https://saintleo-advocate.symplicity.com/index.php" rel="noopener noreferrer" target="_blank">JAMS</a>  
                <a class="dropdown-item" href="https://slu-residence.symplicity.com/index.php/pid648336?#" rel="noopener noreferrer" target="_blank">Residence</a>  
                <a class="dropdown-item" href="https://teamup.com/ks5d76f23008146a4d" rel="noopener noreferrer" target="_blank">Program Calendar</a>  
            </div>  
        </li>  
    </ul>  
</div>
```

```
<!--<div class="dropdown-divider"></div>-->
</div>
</li>
<li class="nav-item">
<!--<a class="nav-link disabled" href="#">Disabled</a> -->
</li>
</ul>
</div>
</nav>

<div class="jumbotron jumbotron-fluid text-center">
<h1 class="display-4">On-Call Quick Access</h1>
<div class = container>
<p class="lead">Need help with a situation? Quickly call the On-Call Resident Assistant Director or Professional Staff using our Quick Access!</p>
<hr>
<div class= row>
<div class="col-md-4 col-xl-4">
<div class="card">
<div class="card-body">
<h2 class="card-title text-center">When To Call Professional Staff</h2>
<p class="card-text text-left">-Suicidal Gesturing<br>-Major Vandalism (Building door is broken, Flood, Fire, Causing harm or safety issues)<br>-Physical or Sexual Assault (including rumors of assault)<br>-Student Death or Injury<br>-Possession of Weapons<br>-Drugs/Paraphernalia<br>-Fire or Flood<br>-Student Transported to the Hospital</p>
<br><hr>
<a href="OnCall.html" class="btn btn-primary">Click to Contact<br>Professional Staff On-Call</a><hr>
```

```
</div>
</div>
</div>

<div class="col-md-4 col-xl-4">
<div class="card">
</div>
</div>

<div class="col-md-4 col-xl-4">
<div class="card">
<div class="card-body">
<h2 class="card-title text-center">When To Call <br>RAD On-
Call</h2>
<p class="card-text text-left">-Pet Violations<br>-Vandalism (Graffiti, Small hole in
a wall, Furniture damaged, Tagging) <br>-Small Roommate Disagreements<br>-Alcohol
Found/Party<br>-Noise Complaints<br>-Cannot find Origin of Smell/Drugs Were NOT
Found<br>-Toilet Won't Stop Flushing<br>-Visitation/Escort Policy (not endangering
residents)</p>
<br><hr>
<a href="OnCall.html" class="btn btn-primary">Click to
Contact<br>Resident Assistant Director On-Call</a><hr>
</div>
</div>
</div>

<div class=container>
<br><h3><i>Please do not hesitate to call us if you need
assistance, we are here to help you!</i></h3><br><br><br><br></div>

<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
```

```
<script src="js/jquery-3.4.1.min.js"></script>

<!-- Include all compiled plugins (below), or include individual files as needed -->
<script src="js/popper.min.js"></script>
<script src="js/bootstrap-4.4.1.js"></script>
</body>

<!-- Footer -->
<footer class="footer text-white" id="wrap">
    <table align="center">
        <col width= "100px">
        <col width= "400px">
        <tr>
            <td></td>
            <td valign="middle" class="text-left">Saint Leo University | Office of Residence Life<br><i>33701 State Road 52 | Saint Leo, FL 33574<br>Main Office: <a href="tel:3525888268" class="text-white">(352) 588-8268</a></i><br>"Love To Live Here"</td>
        </tr>
    </table>
</footer>
</html>
```

Appendix J - Meeting Notes with TI3

Notes during UTS Meeting:

- Maxwell K, Ryan R, Jared D, Sean V, Gregory Hodges (UTS) in attendance
- MK shows the website of the application and the mobile version. MK told the purpose of the application. Showed the issues with current RA website with the security aspect.
- MK expressed the need to create the accounts and integrate with Okta
- Jay Gormley (application expert) joined the call
- MK re-explains the website and tools to JG
- Website is uploaded to GitHub, but coded in Node.js. Brand new development
- JG=Two ways integrating= SAML 2.0, AIS
- JG=Open ID and using .NET
- Problem integrating all of apps into one app
- Trying to implement authentication
- GH= creating a trust relationship, creating an active directory, pass email credentials, need to decide whether first.last or even email. Must be decided in the back end.
- MK= Do we need to provide a database?
- GH=manually, CSV files
- MK=can we do that through email?
- GH=need to have a password
- JG=create a security group, and make a token within the group
- JG=work with UTS to create a group, and trust

The screenshot shows a Zoom meeting interface with four participants: Ryan Roper, Jared DeWinne, gregory.hodges62, and Jay Gormley. A green bar at the top indicates "You are viewing gregory.hodges62's camera". Below the video, there is a "View Options" dropdown menu.

The main content area displays the "SAM Settings" configuration page. It includes sections for "Single sign-on URL", "Audience URL (SP Entity ID)", "Default RelayState", "Name ID Format", "Application Username", and "Update application username on". There are also sections for "Attribute Statements (Optional)" and "Claims Based". On the right side, there are informational boxes about what SAML is, where to find the info it needs, and how to generate one.

Below the settings page, a browser window is open to "https://helpdesk.uvm.edu/saml/sam/settings", showing the same SAML Settings configuration page. The browser tabs include "UVM University - Appcenter", "https://helpdesk.uvm.edu/saml/sam/settings", and "https://zoom.us/j/6103196366?pwd=VWZkMzFhR2dJZGJhUTBjZ0tXZ0l3QT09".

- JD= Mongo DB is the database being used

The screenshot shows a Zoom meeting interface with four participants: Ryan Roper, Jared DeWinne, gregory.hodges62, and Jay Gormley. A green bar at the top indicates "You are viewing gregory.hodges62's camera". Below the video, there is a "View Options" dropdown menu.

The main content area displays a screenshot of the TOPlan application interface. The left sidebar shows "Welcome Gregory Hodges" and lists "Tasks", "Modules", "Inboxes", and "Offices". The "Tasks" section shows a list of items like "Second Line Requests", "Process", "Simple Changes", "Change Activities", and "Authorization Requests". The "Inboxes" section shows "Implementation Team" with items like "Incidents", "Simple Changes", "Change Activities", and "Offices". The "Offices" section shows "Incidents", "Problems and Issues", "Simple Changes", and "All Changes". A large central pane shows a detailed view of a task or item, with a scroll bar visible.

Below the application screenshot, a browser window is open to "https://helpdesk.uvm.edu/saml/sam/settings", showing the same SAML Settings configuration page. The browser tabs include "UVM University - Appcenter", "https://helpdesk.uvm.edu/saml/sam/settings", and "https://zoom.us/j/6103196366?pwd=VWZkMzFhR2dJZGJhUTBjZ0tXZ0l3QT09".

SAML information that is passed back and forth

- Make sure everything documented
- Supporting this application for long term.

Appendix K

Below is the letter from TI3 and Student Affairs to Dr. Van Nguyen highlighting the University's intent to use our web application and implement it into their environment:

Student Affairs



University Campus
Post Office Box 6665
Saint Leo, FL 33574-6665
Office: (352) 588-8992
Fax: (352) 588-6401

April 16, 2020

Dr. Manh Van Ngyuen,

We are writing to inform you that as part of a capstone project that Maxwell Kizewski, Jared Dewinne, Ryan Roper have been working on for you in their COM497 course the Residence Life Duty application that they have been developing will be put into production in the future. Unfortunately due to time constraints and additional requirements on the University's end this application will have to be migrated into a different environment over the summer for deployment so that the content that the students developed can be sustained, correctly integrated, and supported by the University' TI3 Office. The work that they did addressed several security points, once content is migrated into our environment will enhance the security of directory information and access for the Residence Life office. If you have any questions on this please feel free to contact us, Manny or Sean, at your convenience.

Sincerely,

A handwritten signature in black ink, appearing to read "El Rody".

Manny Rodriguez
CTO
emmanuel.rodriguez03@saintleo.edu

A handwritten signature in black ink, appearing to read "NN. Vt".

Sean VanGuilder
Executive Director, Campus Life/Student Affairs
sean.vanguard@ saintleo.edu

Appendix L - Okta Integration

In the Okta Integration, a few things were required. One was the static IP address. This is due to the requirement of Okta needing a constant source to send authentication to and from. The second was a developer account that had Admin permissions for its relevant sites. Below are the pictures of the SAML2.0 Plugin (miniOrange SSO) within Wordpress, and the Admin account within Okta.

The screenshot shows the Okta Application Management interface. At the top, there's a navigation bar with 'Classic UI', a search bar, and links for 'J. DeWinne', 'RAAA - Saint Leo University-dev-793660', 'Help and Support', 'Sign out', 'Get Started' (with a red notification badge '4'), and 'Upgrade'. Below the navigation is a blue header bar with 'okta' logo, 'Dashboard', 'Directory', 'Applications', 'Security', 'Workflow', 'Reports', 'Settings', 'Upgrade', and 'My Apps' (with a plus icon).

The main content area shows the 'RAAA_WP' application details. It has a gear icon and a 'Edit' button. Below it are tabs for 'Active' and 'View Logs'. The 'General' tab is selected, showing 'Sign On' (which is underlined in green), 'Mobile', 'Import', and 'Assignments'.

On the right side, there's a sidebar with 'About' and 'Application Username' sections. The 'About' section describes SAML 2.0 streamlining user experience by not requiring users to know their credentials. The 'Application Username' section says to choose a format or enter it manually if none is selected.

The 'Sign On Methods' section shows 'SAML 2.0' selected. Below it is a 'Default Relay State' input field.

How to Configure SAML 2.0 for RAAA_WP Application

The following is needed to configure RAAA_WP

- ① Identity Provider Single Sign-On URL:

```
https://dev-793660.okta.com/app/raaaainteouniversitydev793660_raaaawp_1/exk51vy4d6wpfc4su4x6/sso/saml
```

- ② Identity Provider Issuer:

```
http://www.okta.com/exk51vy4d6wpfc4su4x6
```

- ③ X.509 Certificate:

```
-----BEGIN CERTIFICATE-----
MIIDpCCAcgAwIBAgJGAXEYckXMA0cCSq0SlBDQEBwUAM1GSMQs0CQYDQQEewJVUzETMBEG
A1UECAwKQ2FsaWZvcm5pTEMBQQAUEBwNUl2fUIEZyIW5jaXNjczENMAs0A1UECgwT2k0YTEU
MB1GAIUECwU1NPUHJvdnl1ZX1xEzABgNVAMCnR1d18307M29JaXh0AaBgkqhki9w0B0QEW
DWUzJwN9a2l1Y55jB2oWhnchMjAwMz12KjAwMz2hWnchMjAwMz12MjAwN0MzwjC8kJELMAgA1UE
BHMBVXkdEzARbgNVBaGCKNbd1mb3JuawExFzJAUBgNVBaMDNh0iB3cmFuVz1zT28xDTALBgNV
BApMBE9r3GEfxfDASBgVbAeNC1NTT1Byb3ZpZGVyMRMwE0YDVQDQDApexZXYLNsxzNjYwMrwvGyJ
KoZ21nuNa0wRFptbmhvcGhsd3tuy29hM1R1JANBgkqhkiGw0BQAEFAAACQAM1IBcKACQFA
1un37McGawewic1ZUTN01tp0Lkuvhyl0Zcp1MsK2tJgr7Df2wfFde74aywCgUBPZ5cQ0H
oAx02Riz6HLFB4ETAWLxao1rwQseefUTJcoXM371/UWrC2RghhINh-WgNTg04109hymdfG1h
2U43TRfNrr1v0X02b0ly0paf71scKfJT7dn9gmDyrUR6q2s-0d4W1Y1BEf1QjQmK23zMuWSR
IV5QDQ6ndrpzD2-n+nb,60Ljcam67wmPT1aE2duoxemc1N0mSo1KvbgR/C2U1B45Vj+0+eq11jp
t4H4m0kBTN7bLSFxeinawv307rJeqA9WwQ1QDAOABMAGC5q0Slb3DQEbwUAAA1BAAQhW+Tka
qBRwocb95MfJ/ BX0v0eUFpxzV3Ojsgyvpu1s7QNj2420MZ2ZIBauBvTzJsuJwFEP3ak+8YmD
RECRFfXgOY1/plqV+9fw0C2P9h1wOb-n/8V8s0engfisxd1yc/4cdlDUQW93HjzKnrDy
YZ1isWh3SX9f877LB8kY8w1sdTlm/gLA1sbh887+7B9pIR1d251i6xrre4t2+/+1W7w14BtgxQ
VHAfxuUSF1zthHo272o0PaNj4V60APEW0Nr1QPj2g0t6yXcYhViJacezdmEeu1mBKVS3hwrc6
UA90D09VHK/RlxNotrwf0rwicZTERu
-----END CERTIFICATE-----
```

miniOrange SSO using SAML 2.0 [Upgrade Now](#) [FAQs](#) [Ask questions on our forum](#) [Restart Plugin Tour](#)

[Service Provider Setup](#) [Service Provider Metadata](#) [Attribute/Role Mapping](#) [Premium plans](#) [Redirection & SSO Links](#) [Add-Ons](#) [Demo Request](#) [Account Setup](#)

Configure Service Provider

Select your Identity Provider from the list below, and you can find the link to the guide for setting up SAML below. Please contact us if you don't find your IDP in the list.

Select configuration guide for your Identity Provider : ADFS [Link to guide](#)

Identity Provider Name *: Okta

IDP Entity ID or Issuer *: <http://www.okta.com/exk5lv4d6WpfC4su4x6>
Note : You can find the EntityID in Your IdP-Metadata XML file enclosed in `<EntityDescriptor>` tag having attribute as `entityID`

SAML Login URL *: https://dev-793660.okta.com/app/raasaintleouniversitydev793660.raawap_1/exk5lv4d6WpfC4su4x6/sso/saml
Note : You can find the SAML Login URL in Your IdP-Metadata XML file enclosed in `<SingleSignOnService>` tag (Binding type: HTTP-Redirect)

X.509 Certificate *:

```
-----BEGIN CERTIFICATE-----
MIIDpDCCAOgAwIBAgIgAXEYkkYMA0GCSqGSIb3DQEBCwUAMIGSMQswCQYDVQQG
EwJVUzETMBEGA1UECAwKQ2FsaWZvcn5pYTEWMBQGA1UEBwwNjU2FuEzYwM5jaXNj
bzENMAsGA1UECgwET21OYTEUMBIGA1UECwvLU1NPUEHJvdmlkZXIxEzARBgNVBAMM
CmRldj03OTM2NjAxIDAQBgkqhkiG9w0BCQEWDVuLzrn9Ab2lOY5jb20whNMjAw
MzI2MjAwMz2MjWhcMz2MjAwNDMzWjCBkJELMAkGA1UEBhMCVVmxEzARBgNV
-----
```

NOTE: Format of the certificate:
`-----BEGIN CERTIFICATE-----`
`XXXXXXXXXXXXXXXXXXXXXX`
`-----END CERTIFICATE-----`

Character encoding : NOTE: Uses iconv encoding to convert X509 certificate into correct encoding.

[Save](#) [Test configuration](#) [Export Plugin Configuration](#)

Feature Request/Contact Us
 Need any help? We can help you with configuring your Identity Provider. Just send us a query and we will get back to you soon.
If you have any query related to the Licensing Plans or add-ons, please let us know, we will assist you in choosing the right plan as per your requirement.

Enter your email
 +1
 Write your query here
 Send plugin configuration with the query

[Submit](#)

Keep configuration Intact
 Enabling this would keep your settings intact when plugin is uninstalled
 Please enable this option when you are updating to a Premium version.

Appendix M - SAML 2.0 SSO Config File

```
<?php
/*
Plugin Name: miniOrange SSO using SAML 2.0
Plugin URI: http://miniorange.com/
Description: miniOrange SAML 2.0 SSO enables user to perform Single Sign On with any
SAML 2.0 enabled Identity Provider.
Version: 4.8.87
Author: miniOrange
Author URI: http://miniorange.com/
*/
```

```
include_once dirname( __FILE__ ) . '/mo_login_saml_sso_widget.php';
require( 'mo-saml-class-customer.php' );
require( 'mo_saml_settings_page.php' );
require( 'MetadataReader.php' );
require( "feedback_form.php" );
require_once "PointersManager.php";
include_once 'Utilities.php';

require_once dirname(__FILE__).'/includes/lib/MoSAMLPointer.php';

class saml_mo_login {

    function __construct() {
        add_action( 'admin_menu', array( $this, 'miniorange_sso_menu' ) );
        add_action( 'admin_init', array( $this,
'miniorange_login_widget_saml_save_settings' ) );
        add_action( 'admin_enqueue_scripts', array( $this, 'plugin_settings_style' ) );
        register_deactivation_hook( __FILE__, array( $this, 'mo_saml_deactivate' ) );
    }
}
```

```

        add_action( 'admin_enqueue_scripts', array( $this, 'plugin_settings_script' ) );
        remove_action( 'admin_notices', array( $this, 'mo_saml_success_message' ) );
        remove_action( 'admin_notices', array( $this, 'mo_saml_error_message' ) );
        add_action( 'wp_authenticate', array( $this, 'mo_saml_authenticate' ) );
        add_action( 'admin_footer', array( $this, 'feedback_request' ) );

        add_action( 'plugin_action_links_' . plugin_basename( __FILE__ ),
array($this,'mo_saml_plugin_action_links') );
register_activation_hook(__FILE__,array($this,'plugin_activate'));
add_action( 'wp_ajax_skip_entire_plugin_tour', array($this, 'handle_skip_entire_plugin'));

}

function handle_skip_entire_plugin(){
    update_option('mo_is_new_user',1);
    $uid = get_current_user_id();
    $array_dissmised_pointers = explode( ',', (string) get_user_meta( $uid,
'dismissed_wp_pointers', TRUE ) );
    $array_dissmised_pointers = array_diff($array_dissmised_pointers,$this->merge_all_pointers());
    update_option('plugin_wise_tour_initiated',true);
    $array_dissmised_pointers = array_merge($array_dissmised_pointers,$this->merge_all_pointers());
    update_user_meta($uid,'dismissed_wp_pointers',implode(",",$array_dissmised_pointers));
    exit;
}

function merge_all_pointers(){

    $array = array();
    return array_merge($array, mo_saml_options_enum_pointersMoSAML::$ATTRIBUTE_MAPPING,

```

```
    mo_saml_options_enum_pointersMoSAML::$DEFAULT_SKIP,
    mo_saml_options_enum_pointersMoSAML::$IDENTITY_PROVIDER,
    mo_saml_options_enum_pointersMoSAML::$REDIRECTION_LINK,
    mo_saml_options_enum_pointersMoSAML::$SERVICE_PROVIDER);

}

function feedback_request() {

    mo_saml_display_saml_feedback_form();
}

function mo_login_widget_saml_options() {
    global $wpdb;

    mo_saml_register_saml_sso();
}

function mo_saml_success_message() {
    $class = "error";
    $message = get_option( 'mo_saml_message' );
    echo "<div class=\"" . $class . "\"><p>" . $message . "</p></div>";
}

function mo_saml_error_message() {
    $class = "updated";
    $message = get_option( 'mo_saml_message' );
    echo "<div class=\"" . $class . "\"><p>" . $message . "</p></div>";
}

public function mo_saml_deactivate(){

if(mo_saml_is_customer_registered_saml(false))
```

```
    return;
    if(!mo_saml_is_curl_installed())
        return;
    $customer = new Customersaml();
    $customer->enable_guest_audit(false);
    wp_redirect('plugins.php');

}

public function enable_guest(){

    if(mo_saml_is_customer_registered_saml(false))
        return;

    if(mo_saml_is_curl_installed()){
        $customer = new Customersaml();
        $customer->enable_guest_audit(true);
    }
    return;
}

public function mo_saml_remove_account() {
    if( ! is_multisite() ) {
        //delete all customer related key-value pairs
        delete_option( 'mo_saml_host_name' );
        delete_option( 'mo_saml_new_registration' );
        delete_option( 'mo_saml_admin_phone' );
        delete_option( 'mo_saml_admin_password' );
        delete_option( 'mo_saml_verify_customer' );
        delete_option( 'mo_saml_admin_customer_key' );
        delete_option( 'mo_saml_admin_api_key' );
        delete_option( 'mo_saml_customer_token' );
        delete_option('mo_saml_admin_email');
        delete_option( 'mo_saml_message' );
    }
}
```

```
        delete_option( 'mo_saml_registration_status' );
        delete_option( 'mo_saml_idp_config_complete' );
        delete_option( 'mo_saml_transactionId' );
        delete_option( 'mo_proxy_host' );
        delete_option( 'mo_proxy_username' );
        delete_option( 'mo_proxy_port' );
        delete_option( 'mo_proxy_password' );
        delete_option( 'mo_saml_show_mo_idp_message' );

    } else {
    global $wpdb;
    $blog_ids      = $wpdb->get_col( "SELECT blog_id FROM $wpdb->blogs" );
    $original_blog_id = get_current_blog_id();

    foreach ( $blog_ids as $blog_id ) {
        switch_to_blog( $blog_id );
        //delete all your options
        //E.g: delete_option( {option name} );
        delete_option( 'mo_saml_host_name' );
        delete_option( 'mo_saml_new_registration' );
        delete_option( 'mo_saml_admin_phone' );
        delete_option( 'mo_saml_admin_password' );
        delete_option( 'mo_saml_verify_customer' );
        delete_option( 'mo_saml_admin_customer_key' );
        delete_option( 'mo_saml_admin_api_key' );
        delete_option( 'mo_saml_customer_token' );
        delete_option( 'mo_saml_message' );
        delete_option( 'mo_saml_registration_status' );
        delete_option( 'mo_saml_idp_config_complete' );
        delete_option( 'mo_saml_transactionId' );
        delete_option( 'mo_saml_show_mo_idp_message' );
```

```
        delete_option('mo_saml_admin_email');
        delete_option('mo_is_new_user');
    }
    switch_to_blog( $original_blog_id );
}
}

function plugin_settings_style( $page ) {
    if ( $page != 'toplevel_page_mo_saml_settings' && !(isset($_REQUEST['page']))
&& $_REQUEST['page'] == 'mo_saml_licensing') {
        return;
    }
    if((isset($_REQUEST['tab']) && $_REQUEST['tab'] == 'licensing') ||
(isset($_REQUEST['page'])) && $_REQUEST['page'] == 'mo_saml_licensing')){
        wp_enqueue_style( 'mo_saml_bootstrap_css', plugins_url( 'includes/css/bootstrap/
bootstrap.min.css', __FILE__ ) );
    }

    wp_enqueue_style( 'mo_saml_admin_settings_jquery_style', plugins_url( 'includes/css/
jquery.ui.css', __FILE__ ) );
    wp_enqueue_style( 'mo_saml_admin_settings_style', plugins_url( 'includes/css/
style_settings.min.css', __FILE__ ), array(), '4.8.82', 'all' );
    wp_enqueue_style( 'mo_saml_admin_settings_phone_style', plugins_url( 'includes/css/
phone.css', __FILE__ ) );
    wp_enqueue_style( 'mo_saml_wpb-fa', plugins_url( 'includes/css/all.min.css', __FILE__ ) );
    $file = plugin_dir_path( __FILE__ ) . 'pointers.php';
    $manager = new MoSAMLPointersManager( $file, '4.8.52',
'custom_admin_pointers' );
    $manager->parse();
    $pointers = $manager->filter( $page );
    if ( empty( $pointers ) ) {
        return;
    }
```

```
wp_enqueue_style( 'wp-pointer' );
$js_url = plugins_url( 'includes\js\pointers.js', __FILE__ );
wp_enqueue_script( 'custom_admin_pointers', $js_url, array('wp-pointer'), NULL, TRUE );
$data = array(
    'close_label' => __('Close'),
    'next_label' => __('Next'),
    'pointers' => $pointers
);
wp_localize_script( 'custom_admin_pointers', 'MOAdminPointers', $data );

}

function plugin_settings_script( $page ) {

    if ( $page != 'toplevel_page_mo_saml_settings' && !isset($_REQUEST['page'])
&& $_REQUEST['page'] == 'mo_saml_licensing') {
        return;
    }
    wp_register_script( 'rml-script', plugins_url( 'includes/js/skip_tour.js', __FILE__ ),
array('jquery'), null, true );
    wp_localize_script( 'rml-script', 'readmelater_ajax', array( 'ajax_url' => admin_url('admin-
ajax.php')) );

    wp_enqueue_script( 'jquery' );
    wp_enqueue_script( 'mo_saml_admin_settings_script', plugins_url( 'includes/js/
settings.js', __FILE__ ) );
    wp_enqueue_script( 'mo_saml_admin_settings_phone_script',
plugins_url( 'includes/js/phone.js', __FILE__ ) );

    if((isset($_REQUEST['tab']) && $_REQUEST['tab'] == 'licensing') ||
(isset($_REQUEST['page']) && $_REQUEST['page'] == 'mo_saml_licensing')){
```

```
        wp_enqueue_script( 'mo_saml_modernizr_script', plugins_url( 'includes/js/modernizr.js',
__FILE__ ) );
        wp_enqueue_script( 'mo_samlPopover_script', plugins_url( 'includes/js/bootstrap/
popper.min.js', __FILE__ ) );
        wp_enqueue_script( 'mo_saml_bootstrap_script', plugins_url( 'includes/js/bootstrap/
bootstrap.min.js', __FILE__ ) );
    }

}
```

```
public function plugin_activate(){
    if(is_multisite()){
        global $wpdb;
        $blog_ids = $wpdb->get_col("SELECT blog_id FROM $wpdb->blogs");
        $original_blog_id = get_current_blog_id();

        foreach($blog_ids as $blog_id){
            switch_to_blog($blog_id);
            update_option('mo_saml_guest_log',true);
            update_option('mo_saml_guest_enabled',true);
            update_option( 'mo_saml_free_version', 1 );

            $uid = get_current_user_id();
            $array_dissmised_pointers = explode( ',', (string) get_user_meta( $uid,
'dismissed_wp_pointers', TRUE ) );
            $array_dissmised_pointers =
array_merge($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::
$DEFAULT);
            $array_dissmised_pointers =
array_merge($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::
$DEFAULT_SKIP);
```

```
        update_user_meta($uid,'dismissed_wp_pointers',implode(",",
$array_dissmised_pointers));

    }

    switch_to_blog($original_blog_id);

} else {

    update_option('mo_saml_guest_log',true);
    update_option('mo_saml_guest_enabled',true);
    update_option( 'mo_saml_free_version', 1 );

$uid = get_current_user_id();

$array_dissmised_pointers = explode( ',', (string) get_user_meta( $uid,
'dismissed_wp_pointers', TRUE ) );

$array_dissmised_pointers =
array_merge($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$DEFAULT);

$array_dissmised_pointers =
array_merge($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$DEFAULT);

        update_user_meta($uid,'dismissed_wp_pointers',implode(",",
$array_dissmised_pointers));

    }

    update_option('mo_plugin_do_activation_redirect', true);
    $this->enable_guest();
}

static function mo_check_option_admin_referer($option_name){

    return (isset($_POST['option']) and $_POST['option']===$option_name and
check_admin_referer($option_name));

}

function miniorange_login_widget_saml_save_settings() {

    if (get_option('mo_plugin_do_activation_redirect')) {
```

```
delete_option('mo_plugin_do_activation_redirect');

if(!isset($_GET['activate-multi']))
{
    wp_redirect(admin_url() . 'admin.php?
page=mo_saml_settings');
    exit;
}

if( current_user_can( 'manage_options' ) ) {

    if(self::mo_check_option_admin_referer('dismiss_pointers'))
{

    $uid = get_current_user_id();
    $array_dissmised_pointers = explode( ',', (string) get_user_meta( $uid,
'dismissed_wp_pointers', TRUE ) );
    if( isset( $_GET['tab'] ) ) {
        $active_tab = $_GET['tab'];
        if($active_tab == 'save')

$array_dissmised_pointers=array_merge($array_dissmised_pointers,mo_saml_options_enum_po
intersMoSAML::$SERVICE_PROVIDER);
        elseif($active_tab == 'config')

$array_dissmised_pointers=array_merge($array_dissmised_pointers,mo_saml_options_enum_po
intersMoSAML::$IDENTITY_PROVIDER);
        elseif ($active_tab == 'opt')

$array_dissmised_pointers=array_merge($array_dissmised_pointers,mo_saml_options_enum_po
intersMoSAML::$ATTRIBUTE_MAPPING);
        elseif ($active_tab == 'general')
```

```
$array_dissmised_pointers=array_merge($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$REDIRECTION_LINK);

}else {

$array_dissmised_pointers=array_merge($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$DEFAULT,mo_saml_options_enum_pointersMoSAML::$SERVICE_PROVIDER,mo_saml_options_enum_pointersMoSAML::$ATTRIBUTE_MAPPING,
    mo_saml_options_enum_pointersMoSAML::$IDENTITY_PROVIDER,
    mo_saml_options_enum_pointersMoSAML::$REDIRECTION_LINK);
}

update_user_meta($uid,'dismissed_wp_pointers',implode(",",
$array_dissmised_pointers));
return;

}

if(self::mo_check_option_admin_referer('restart_plugin_tour'))
{

update_option('mo_is_new_user',1);

$uid = get_current_user_id();
$array_dissmised_pointers = explode( ',', (string) get_user_meta( $uid,
'dismissed_wp_pointers', TRUE ) );

$array_dissmised_pointers=array_diff($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$DEFAULT);
```

```
        update_user_meta($uid,'dismissed_wp_pointers',implode(",",
$array_dissmised_pointers));

        update_option('plugin_wise_tour_initiated',true);

        $request_uri = $_SERVER['REQUEST_URI'];

        $redirect_array=explode('&',htmlentities($request_uri));

        $redirect= $redirect_array[0];

        header("Location: ".$redirect);

        return;

    }

if(self::mo_check_option_admin_referer('skip_plugin_tour'))
{

    update_option('mo_is_new_user',1);

    $uid = get_current_user_id();

    $array_dissmised_pointers = explode( ',', (string) get_user_meta( $uid,
'dismissed_wp_pointers', TRUE ) );

    $array_dissmised_pointers =
array_diff($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::
$DEFAULT);

    update_option('plugin_wise_tour_initiated',true);

    $array_dissmised_pointers =
array_merge($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::
$DEFAULT_SKIP);

    update_user_meta($uid,'dismissed_wp_pointers',implode(",",
$array_dissmised_pointers));

    $redirect_array =explode('&',htmlentities($_SERVER['REQUEST_URI']));

    $redirect=$redirect_array[0];
```

```
header("Location: ".$redirect);
return;

}

if(self::mo_check_option_admin_referer("clear_attrs_list")){
    delete_option("mo_saml_test_config_attrs");
    update_option('mo_saml_message','List of attributes cleared');
    $this->mo_saml_show_success_message();
}

if ( self::mo_check_option_admin_referer("clear_pointers") ) {

    $uid = get_current_user_id();
    $array_dissmised_pointers = explode( ',', (string) get_user_meta( $uid,
'dismissed_wp_pointers', TRUE ) );

switch ($_POST['button_name']){
    case mo_saml_options_tab_names::Entire_plugin_tour:
        $array_dissmised_pointers =
array_diff($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$DEFAULT);
        break;
    case mo_saml_options_tab_names::Attribute_role_mapping:
        $array_dissmised_pointers =
array_diff($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$ATTRIBUTE_MAPPING);
        break;
    case mo_saml_options_tab_names::Identity_provider_settting:
```

```

$array_dissmised_pointers =
array_diff($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$IDENTITY_PROVIDER);
break;
case mo_saml_options_tab_names::Redirection_sso_links:
$array_dissmised_pointers =
array_diff($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$REDIRECTION_LINK);
break;
case mo_saml_options_tab_names::Service_provider_settings:
update_option('service_provider_setup_tour_initiated',true);
$array_dissmised_pointers =
array_diff($array_dissmised_pointers,mo_saml_options_enum_pointersMoSAML::$SERVICE_PROVIDER);
break;

}

update_user_meta($uid,'dismissed_wp_pointers',implode(",",
$array_dissmised_pointers));
return;
}
if( isset( $_POST['option'] ) and $_POST['option'] ==
"mo_saml_mo_idp_message" ) {
update_option( 'mo_saml_show_mo_idp_message', 1 );

return;
}
if( self::mo_check_option_admin_referer("change_miniorange")){
self::mo_saml_remove_account();
update_option('mo_saml_guest_enabled',true);
//update_option( 'mo_saml_message', 'Logged out of miniOrange account' );

```

```
//$this->mo_saml_show_success_message();
return;
}

if
( self::mo_check_option_admin_referer("login_widget_saml_save_settings")) {
    if ( ! mo_saml_is_curl_installed() ) {
        update_option( 'mo_saml_message', 'ERROR: <a href="http://php.net/manual/en/curl.installation.php" target="_blank">PHP cURL extension</a> is not installed or disabled. Save Identity Provider Configuration failed.' );
        $this->mo_saml_show_error_message();

    return;
}

//validation and sanitization
$saml_identity_name = "";
$saml_login_url     = "";
$saml_issuer         = "";
$saml_x509_certificate = "";
if ( $this->mo_saml_check_empty_or_null( $_POST['saml_identity_name'] ) || $this->mo_saml_check_empty_or_null( $_POST['saml_login_url'] ) || $this->mo_saml_check_empty_or_null( $_POST['saml_issuer'] ) ) {
    update_option( 'mo_saml_message', 'All the fields are required. Please enter valid entries.' );
    $this->mo_saml_show_error_message();

    return;
} else if ( ! preg_match( "/^\\w*$/", $_POST['saml_identity_name'] ) ) {
}
```

```
        update_option( 'mo_saml_message', 'Please match the
requested format for Identity Provider Name. Only alphabets, numbers and underscore is
allowed.' );
        $this->mo_saml_show_error_message();

        return;
    } else {
        $saml_identity_name =
sanitize_text_field(trim( $_POST['saml_identity_name'] ));
        $saml_login_url =
htmlspecialchars(trim( $_POST['saml_login_url'] ));
        $saml_issuer =
htmlspecialchars(trim( $_POST['saml_issuer'] ));
        $saml_identity_provider_guide_name =
htmlspecialchars(trim($_POST['saml_identity_provider_guide_name']));
        $saml_x509_certificate =
$_POST['saml_x509_certificate'];

    }

        update_option( 'saml_identity_name', $saml_identity_name );
        update_option( 'saml_login_url', $saml_login_url );
        update_option( 'saml_issuer', $saml_issuer );
        update_option( 'saml_identity_provider_guide_name',
$saml_identity_provider_guide_name);

        if ( isset( $_POST['saml_response_signed'] ) ) {
            update_option( 'saml_response_signed', 'checked' );
        } else {
            update_option( 'saml_response_signed', 'Yes' );
        }
    }
}
```

```
foreach ( $saml_x509_certificate as $key => $value ) {
    if ( empty( $value ) ) {
        unset( $saml_x509_certificate[ $key ] );
    } else {
        $saml_x509_certificate[ $key ] =
            Utilities::sanitize_certificate( $value );

        if ( ! @openssl_x509_read( $saml_x509_certificate[
            $key ] ) ) {
            update_option( 'mo_saml_message', 'Invalid
certificate: Please provide a valid certificate.' );
            $this->mo_saml_show_error_message();
            delete_option( 'saml_x509_certificate' );

            return;
        }
    }
}

if ( empty( $saml_x509_certificate ) ) {
    update_option( "mo_saml_message", 'Invalid Certificate:
Please provide a certificate' );
    $this->mo_saml_show_error_message();

    return;
}
update_option( 'saml_x509_certificate',
    maybe_serialize( $saml_x509_certificate ) );
if ( isset( $_POST['saml_assertion_signed'] ) ) {
    update_option( 'saml_assertion_signed', 'checked' );
}
else {
    update_option( 'saml_assertion_signed', 'Yes' );
}
```

```
if(array_key_exists('enable_icconv', $_POST))
    update_option('mo_saml_encoding_enabled', 'checked');
else
    update_option('mo_saml_encoding_enabled', '');

        update_option( 'mo_saml_message', 'Identity Provider
details saved successfully.' );
        $this->mo_saml_show_success_message();

    }

//Update SP Entity ID

if(self::mo_check_option_admin_referer('mo_saml_update_idp_settings_option')){
    if(isset($_POST['mo_saml_sp_entity_id'])) {
        $sp_entity_id =
sanitize_text_field($_POST['mo_saml_sp_entity_id']);
        update_option('mo_saml_sp_entity_id', $sp_entity_id);
    }
    update_option('mo_saml_message', 'Settings updated
successfully.');
    $this->mo_saml_show_success_message();
}

//Save Attribute Mapping
if
(self::mo_check_option_admin_referer("login_widget_saml_attribute_mapping") ) {

    if( ! mo_saml_is_curl_installed() ) {
        update_option( 'mo_saml_message', 'ERROR: <a
href="http://php.net/manual/en/curl.installation.php" target="_blank">PHP cURL extension</a>
is not installed or disabled. Save Attribute Mapping failed.' );
        $this->mo_saml_show_error_message();
}
```

```
        return;
    }

    update_option( 'mo_saml_message', 'Attribute Mapping details
saved successfully' );
    $this->mo_saml_show_success_message();

}

//Save Role Mapping
if
(self::mo_check_option_admin_referer("login_widget_saml_role_mapping")) {

    if( ! mo_saml_is_curl_installed() ) {
        update_option( 'mo_saml_message', 'ERROR: <a
href="http://php.net/manual/en/curl.installation.php" target="_blank">PHP cURL extension</a>
is not installed or disabled. Save Role Mapping failed.' );
        $this->mo_saml_show_error_message();

        return;
    }

    update_option( 'saml_am_default_user_role',
htmlspecialchars($_POST['saml_am_default_user_role']) );

    update_option( 'mo_saml_message', 'Role Mapping details saved
successfully.' );
    $this->mo_saml_show_success_message();
}

if(self::mo_check_option_admin_referer("mo_saml_demo_request_option")){

```

```
if(isset($_POST['mo_saml_demo_email']))
    $demo_email =
htmlspecialchars($_POST['mo_saml_demo_email']);

if(empty($demo_email))
    $demo_email = get_option('mo_saml_admin_email');

if(isset($_POST['mo_saml_demo_plan']))
    $demo_plan =
htmlspecialchars($_POST['mo_saml_demo_plan']);

if(isset($_POST['mo_saml_demo_description']))
    $demo_description =
htmlspecialchars($_POST['mo_saml_demo_description']);

$license_plans = array(
    'standard'          => 'WP SAML SSO Standard Plan',
    'premium'           => 'WP SAML SSO Premium Plan',
    'premium-multisite' => 'WP SAML SSO Premium Multisite Plan',
    'enterprise'        => 'WP SAML SSO Enterprise Plan',
    'enterprise-multisite' => 'WP SAML SSO Enterprise Multisite Plan',
    'enterprise-multiple-idp' => 'WP SAML SSO Enterprise Multiple-IDP Plan',
    'help'               => 'Not Sure'
);
if(isset($license_plans[$demo_plan]))
    $demo_plan = $license_plans[$demo_plan];
$addons = array(
    'page-restriction'      => 'Page Restriction',
    'buddypress-integration' => 'BuddyPress Integration',
    'learndash-integration'  => 'LearnDash Integration',
```

```
'sso-login-audit'      => 'SSO Login Audit',
'attribute-based-redirect' => 'Attribute Based Redirect'
);

$addons_selected = array();
foreach($addons as $key => $value){
    if(isset($_POST[$key]) && $_POST[$key] == "true")
        $addons_selected[$key] = $value;
}

$message = "[Demo For Customer] : " . $demo_email;
if(!empty($demo_plan))
    $message .= " [Selected Plan] : " . $demo_plan;
if(!empty($demo_description))
    $message .= " [Requirements] : " . $demo_description;

if(!empty($addons_selected)){
    $message .= " [Addons] : ";
    foreach($addons_selected as $key => $value){
        $message .= $value;
        if(next($addons_selected))
            $message .= ", ";
    }
}

$user = wp_get_current_user();
$customer = new Customersaml();
$email = get_option( "mo_saml_admin_email" );
if ( $email == " ) {
    $email = $user->user_email;
}
$phone = get_option( 'mo_saml_admin_phone' );
```

```

$submitted = json_decode( $customer->send_email_alert( $email,
$phone, $message, true ), true );

    if ( json_last_error() == JSON_ERROR_NONE ) {
        if ( is_array( $submitted ) && array_key_exists( 'status',
$submitted ) && $submitted['status'] == 'ERROR' ) {
            update_option( 'mo_saml_message',
$submitted['message'] );
            $this->mo_saml_show_error_message();

        }
        else {
            if ( $submitted == false ) {

                update_option( 'mo_saml_message', 'Error
while submitting the query.' );
                $this->mo_saml_show_error_message();
            } else {
                update_option( 'mo_saml_message',
'Thanks! We have received your request and will shortly get in touch with you.');
                $this->mo_saml_show_success_message();
            }
        }
    }

if (self::mo_check_option_admin_referer("saml_upload_metadata")) {
    if ( ! function_exists( 'wp_handle_upload' ) ) {
        require_once(ABSPATH . 'wp-admin/includes/file.php');
    }
    $this->_handle_upload_metadata();
}

```

```
if( self::mo_check_option_admin_referer("mo_saml_register_customer"))
{

    //register the admin to miniOrange
    $user = wp_get_current_user();
    if( ! mo_saml_is_curl_installed() ) {
        update_option( 'mo_saml_message', 'ERROR: <a href='
        "http://php.net/manual/en/curl.installation.php" target=_blank">PHP cURL extension</a>
        is not installed or disabled. Registration failed.' );
        $this->mo_saml_show_error_message();

        return;
    }

    //validation and sanitization
    $email      = "";
    $password   = "";
    $confirmPassword = "";

    if( $this->mo_saml_check_empty_or_null( $_POST['email'] ) ||
        $this->mo_saml_check_empty_or_null( $_POST['password'] ) || $this-
        >mo_saml_check_empty_or_null( $_POST['confirmPassword'] ) ) {

        update_option( 'mo_saml_message', 'Please enter the
        required fields.' );
        $this->mo_saml_show_error_message();

        return;
    } else if( !filter_var($_POST['email'],
FILTER_VALIDATE_EMAIL) ) {
        update_option( 'mo_saml_message', 'Please enter a valid
        email address.' );
        $this->mo_saml_show_error_message();
    }
}
```

```
        return;
    }
    else if($this->
>checkPasswordpattern(strip_tags($_POST['password']))){
    update_option( 'mo_saml_message', 'Minimum 6 characters should be present.
Maximum 15 characters should be present. Only following symbols (!@#.%"^&*-_) should be
present.' );
    $this->mo_saml_show_error_message();
    return;
}
else {

    $email      = sanitize_email( $_POST['email'] );
    $password   =
stripslashes( strip_tags($_POST['password']) );
    $confirmPassword =
stripslashes( strip_tags($_POST['confirmPassword']) );
}

update_option( 'mo_saml_admin_email', $email );

if ( strcmp( $password, $confirmPassword ) == 0 ) {
    update_option( 'mo_saml_admin_password', $password );
    $email  = get_option( 'mo_saml_admin_email' );
    $customer = new CustomerSaml();
    $content = json_decode( $customer->check_customer(),
true );
    if(!is_null($content)){
        if ( strcasecmp( $content['status'],
'CUSTOMER_NOT_FOUND' ) == 0 ) {

            $response = $this->create_customer();
            if(is_array($response) &&
array_key_exists('status', $response) && $response['status'] == 'success'){


```

```
wp_redirect( admin_url( '/
admin.php?page=mo_saml_settings&tab=licensing' ), 301 );
exit;
}
} else {
$response = $this->get_current_customer();
if(is_array($response) &&
array_key_exists('status', $response) && $response['status'] == 'success'){
wp_redirect( admin_url( '/
admin.php?page=mo_saml_settings&tab=licensing' ), 301 );
exit;
}
//$this->mo_saml_show_error_message();
}
}

} else {
update_option( 'mo_saml_message', 'Passwords do not
match.' );
delete_option( 'mo_saml_verify_customer' );
$this->mo_saml_show_error_message();
}

return;
//new starts here

}

else if( self::mo_check_option_admin_referer("mosaml_metadata_download")){
mo_saml_miniorange_generate_metadata(true);
}

if( self::mo_check_option_admin_referer("mo_saml_verify_customer") )
{
//register the admin to miniOrange

if( ! mo_saml_is_curl_installed() ) {
```

```
        update_option( 'mo_saml_message', 'ERROR: <a href="http://php.net/manual/en/curl.installation.php" target="_blank">PHP cURL extension</a> is not installed or disabled. Login failed.' );
        $this->mo_saml_show_error_message();

    return;
}

//validation and sanitization
$email = "";
$password = "";
if ( $this->mo_saml_check_empty_or_null( $_POST['email'] ) ||
$this->mo_saml_check_empty_or_null( $_POST['password'] ) ) {
    update_option( 'mo_saml_message', 'All the fields are required. Please enter valid entries.' );
    $this->mo_saml_show_error_message();

    return;
} else if($this->checkPasswordpattern(strip_tags($_POST['password']))){
    update_option( 'mo_saml_message', 'Minimum 6 characters should be present. Maximum 15 characters should be present. Only following symbols (!@#.%%^&*-_) should be present.' );
    $this->mo_saml_show_error_message();
    return;
} else {
    $email = sanitize_email( $_POST['email'] );
    $password =
stripslashes( strip_tags($_POST['password']) );
}

update_option( 'mo_saml_admin_email', $email );
update_option( 'mo_saml_admin_password', $password );
```

```
$customer = new Customersaml();
$content = $customer->get_customer_key();
if(!is_null($content)){
    $customerKey = json_decode( $content, true );
    if ( json_last_error() == JSON_ERROR_NONE ) {
        update_option( 'mo_saml_admin_customer_key',
$customerKey['id'] );
        update_option( 'mo_saml_admin_api_key',
$customerKey['apiKey'] );
        update_option( 'mo_saml_customer_token',
$customerKey['token'] );
        $certificate = get_option( 'saml_x509_certificate' );
        if ( empty( $certificate ) ) {
            update_option( 'mo_saml_free_version', 1 );
        }
        update_option( 'mo_saml_admin_password', " );
        update_option( 'mo_saml_message', 'Customer
retrieved successfully' );
        update_option( 'mo_saml_registration_status',
'Existing User' );
        delete_option( 'mo_saml_verify_customer' );
        $this->mo_saml_show_success_message();
        //if(is_array($response) &&
array_key_exists('status', $response) && $response['status'] == 'success'){
            wp_redirect( admin_url( '/admin.php?
page=mo_saml_settings&tab=licensing' ), 301 );
            exit;
        //}
    } else {
        update_option( 'mo_saml_message', 'Invalid
username or password. Please try again.' );
        $this->mo_saml_show_error_message();
    }
}
```

```
        update_option( 'mo_saml_admin_password', " );
    }

}

else if

( self::mo_check_option_admin_referer("mo_saml_contact_us_query_option") ) {

    if( ! mo_saml_is_curl_installed() ) {
        update_option( 'mo_saml_message', 'ERROR: <a href="http://php.net/manual/en/curl.installation.php" target="_blank">PHP cURL extension</a> is not installed or disabled. Query submit failed.' );
        $this->mo_saml_show_error_message();

        return;
    }

    // Contact Us query
    $email  = sanitize_email($_POST['mo_saml_contact_us_email']);
    $phone  =
htmlspecialchars($_POST['mo_saml_contact_us_phone']);
    $query  =
htmlspecialchars($_POST['mo_saml_contact_us_query']);

    if(array_key_exists('send_plugin_config',$_POST)===true) {
        $plugin_config_json = mo_saml_miniorange_import_export(true, true);
        $query .= $plugin_config_json;
        delete_option('send_plugin_config');
    }else
        update_option('send_plugin_config','off');

    $customer = new CustomerSaml();
```

```

        if( $this->mo_saml_check_empty_or_null( $email ) || $this-
>mo_saml_check_empty_or_null( $query ) ) {
                update_option( 'mo_saml_message', 'Please fill up Email
and Query fields to submit your query.' );
                $this->mo_saml_show_error_message();
} else if( !filter_var($email, FILTER_VALIDATE_EMAIL) ) {
                update_option( 'mo_saml_message', 'Please enter a valid
email address.' );
                $this->mo_saml_show_error_message();
} else {

        $submited = $customer->submit_contact_us( $email,
$phone, $query );
        if(!is_null($submited)){
                if( $submited == false ) {
                        update_option( 'mo_saml_message', 'Your
query could not be submitted. Please try again.' );
                        $this->mo_saml_show_error_message();
} else {
                        update_option( 'mo_saml_message', 'Thanks
for getting in touch! We shall get back to you shortly.' );
                        $this->mo_saml_show_success_message();
}
}
}
}

else if( self::mo_check_option_admin_referer("mo_saml_go_back") ) {
        update_option( 'mo_saml_registration_status', " );
        update_option( 'mo_saml_verify_customer', " );
        delete_option( 'mo_saml_new_registration' );
        delete_option( 'mo_saml_admin_email' );
        delete_option( 'mo_saml_admin_phone' );
}
}

```

```

        else if ( self::mo_check_option_admin_referer("mo_saml_goto_login") ) {
            delete_option( 'mo_saml_new_registration' );
            update_option( 'mo_saml_verify_customer', 'true' );
        }
        else if
( self::mo_check_option_admin_referer("mo_saml_forgot_password_form_option") ) {
            if ( ! mo_saml_is_curl_installed() ) {
                update_option( 'mo_saml_message', 'ERROR: <a
href="http://php.net/manual/en/curl.installation.php" target="_blank">PHP cURL extension</a>
is not installed or disabled. Resend OTP failed.' );
                $this->mo_saml_show_error_message();
            }
            return;
        }

        $email = get_option( 'mo_saml_admin_email' );

        $customer = new Customersaml();
        $content = json_decode( $customer-
>mo_saml_forgot_password( $email ), true );
        if(!is_null($content)){
            if ( strcasecmp( $content['status'], 'SUCCESS' ) == 0 ) {
                update_option( 'mo_saml_message', 'Your password
has been reset successfully. Please enter the new password sent to ' . $email . ' );
                $this->mo_saml_show_success_message();
            } else {
                update_option( 'mo_saml_message', 'An error
occured while processing your request. Please Try again.' );
                $this->mo_saml_show_error_message();
            }
        }
    }
    /**

```

```
* Added for feedback mechanisms
*/
if ( self::mo_check_option_admin_referer("mo_skip_feedback") ) {
    update_option( 'mo_saml_message', 'Plugin deactivated successfully' );
    $this->mo_saml_show_success_message();
    deactivate_plugins( __FILE__ );

}

if ( self::mo_check_option_admin_referer("mo_feedback") ) {
    $user = wp_get_current_user();

    $message = 'Plugin Deactivated';

    $deactivate_reason_message = array_key_exists( 'query_feedback',
$_POST ) ? htmlspecialchars($_POST['query_feedback']) : false;

    $reply_required = "";
    if(isset($_POST['get_reply']))
        $reply_required = htmlspecialchars($_POST['get_reply']);
    if(empty($reply_required)){
        $reply_required = "don't reply";
        $message.= '<b style="color:red"> &ampnbsp [Reply :'.
$reply_required.'][</b>';
    }else{
        $reply_required = "yes";
        $message.= '[Reply :'.$reply_required.'];
    }

    if(is_multisite())
        $multisite_enabled = 'True';
    else
```

```
$multisite_enabled = 'False';

$message.= ', [Multisite enabled: ' . $multisite_enabled .']';

$message.= ', Feedback :'.$deactivate_reason_message.';

if (isset($_POST['rate']))
    $rate_value = htmlspecialchars($_POST['rate']);

$message.= ', [Rating :'$rate_value.'];

$email = $_POST['query_mail'];
if(!filter_var($email, FILTER_VALIDATE_EMAIL)){
    $email = get_option('mo_saml_admin_email');
    if(empty($email))
        $email = $user->user_email;
}
$phone = get_option( 'mo_saml_admin_phone' );
$feedback_reasons = new Customersaml();
if(!is_null($feedback_reasons)){
    if(!mo_saml_is_curl_installed()){
        deactivate_plugins( __FILE__ );
        wp_redirect('plugins.php');
    } else {
        $submitted = json_decode( $feedback_reasons-
>send_email_alert( $email, $phone, $message ), true );
        if ( json_last_error() == JSON_ERROR_NONE ) {
            if ( is_array( $submitted ) &&
array_key_exists( 'status', $submitted ) && $submitted['status'] == 'ERROR' ) {
                update_option( 'mo_saml_message',
$submitted['message'] );
                $this-
            }
        }
    }
}
>mo_saml_show_error_message();
```

```
        }

        else {

            if( $submited == false ) {

                update_option( 'mo_saml_message', 'Error while submitting the query.' );

                $this->

            >mo_saml_show_error_message();

            }

        }

        deactivate_plugins( __FILE__ );

        update_option( 'mo_saml_message', 'Thank you for the

feedback.' );

        $this->mo_saml_show_success_message();

    }

}

if( self::mo_check_option_admin_referer("molicensingplanselection") ) {

    $env_type = htmlspecialchars($_POST['envtype']);

    $idp_num = htmlspecialchars($_POST['idpnum']);

    $auto_redirect = htmlspecialchars($_POST['autoredirect']);

    $attr_map = htmlspecialchars($_POST['attrmap']);

    $role_map = htmlspecialchars($_POST['rolemap']);

    $slo = htmlspecialchars($_POST['slo']);

    $addon = htmlspecialchars($_POST['addon']);



    $licensing_plan = "Single Site - Standard";

    if($env_type == 'multisite'){

        $licensing_plan = "Multisite Network - Premium";
```

```

if($addon == 'yes' && $idp_num == '1+'){
    $licensing_plan = "Multisite Network - Business";
}else if($idp_num == '1+') {
    $licensing_plan = "Multisite Network - Business";
}else if($addon == 'yes') {
    $licensing_plan = "Multisite Network - Enterprise";
}
}else{
    if($addon == 'yes' || $idp_num == '1+') {
        $licensing_plan = "Single Site - Enterprise";
    }else{
        if($slo == 'yes' || $role_map == 'yes' || $auto_redirect == 'yes' || $attr_map ==
        'yes'){
            $licensing_plan = "Single Site - Premium";
        }
    }
}
update_option('mo_license_plan_from_feedback', $licensing_plan);
update_option( 'mo_saml_license_message', $licensing_plan . ' Plan (highlighted with
red border) will be the best suitable licensing plan as per the SSO details provided by you. If you
still have any concern, please write us at info@xecurify.com.' );
}

function mo_saml_show_error_message() {
    remove_action( 'admin_notices', array( $this, 'mo_saml_error_message' ) );
    add_action( 'admin_notices', array( $this, 'mo_saml_success_message' ) );
}

public function mo_saml_check_empty_or_null( $value ) {
    if( ! isset( $value ) || empty( $value ) ) {
        return true;
    }
}

```

```
    }

    return false;
}

private function mo_saml_show_success_message() {
    remove_action( 'admin_notices', array( $this, 'mo_saml_success_message' ) );
    add_action( 'admin_notices', array( $this, 'mo_saml_error_message' ) );
}

function _handle_upload_metadata() {
    if ( isset( $_FILES['metadata_file'] ) || isset( $_POST['metadata_url'] ) ) {
        if ( ! empty( $_FILES['metadata_file']['tmp_name'] ) ) {
            $file = @file_get_contents( $_FILES['metadata_file']
['tmp_name'] );
        } else {
            if(!mo_saml_is_curl_installed()){
                update_option( 'mo_saml_message', 'PHP cURL extension
is not installed or disabled. Cannot fetch metadata from URL.' );
                $this->mo_saml_show_error_message();
                return;
            }
            $url = filter_var( $_POST['metadata_url'],
FILTER_SANITIZE_URL );
            $response = Utilities::mo_saml_wp_remote_get($url,
array('sslverify'=>false));
            if(!is_null($response))
                $file = $response['body'];
            else
                $file = null;
        }
    }
}
```

```
    if(!is_null($file))
        $this->upload_metadata( $file );
    }

}

function upload_metadata( $file ) {

    $old_error_handler = set_error_handler( array( $this, 'handleXmlError' ) );
    $document      = new DOMDocument();
    $document->loadXML( $file );
    restore_error_handler();
    $first_child = $document->firstChild;
    if( !empty( $first_child ) ) {
        $metadata      = new IDPMetadataReader( $document );
        $identity_providers = $metadata->getIdentityProviders();
        if( !preg_match( "/^\\w*$/",
$_POST['saml_identity_metadata_provider'] ) ) {
            update_option( 'mo_saml_message', 'Please match the requested
format for Identity Provider Name. Only alphabets, numbers and underscore is allowed.' );
            $this->mo_saml_show_error_message();

            return;
        }
        if( empty( $identity_providers ) && !empty( $_FILES['metadata_file']
['tmp_name']) ) {
            update_option( 'mo_saml_message', 'Please provide a valid
metadata file.' );
            $this->mo_saml_show_error_message();

            return;
        }
        if( empty( $identity_providers ) && !empty($_POST['metadata_url']) ) {
```

```

        update_option( 'mo_saml_message', 'Please provide a valid
metadata URL.' );
        $this->mo_saml_show_error_message();

        return;
    }

    foreach ( $identity_providers as $key => $idp ) {
        // $saml_identity_name = preg_match("/^([a-zA-Z0-9-_]+)\\.", $idp-
>getIdpName()) ? $idp->getIdpName() : "";
        $saml_identity_name =
htmlspecialchars($_POST['saml_identity_metadata_provider']);

        $saml_login_url = $idp->getLoginURL( 'HTTP-Redirect' );

        $saml_issuer      = $idp->getEntityID();
        $saml_x509_certificate = $idp->getSigningCertificate();

        update_option( 'saml_identity_name', $saml_identity_name );

        update_option( 'saml_login_url', $saml_login_url );



        update_option( 'saml_issuer', $saml_issuer );
        // certs already sanitized in Metadata Reader
        update_option( 'saml_x509_certificate',
maybe_serialize( $saml_x509_certificate ) );
        break;
    }

    update_option( 'mo_saml_message', 'Identity Provider details saved
successfully.' );
    $this->mo_saml_show_success_message();
} else {
    if(!empty( $_FILES['metadata_file']['tmp_name']))
```

```
{  
    update_option( 'mo_saml_message', 'Please provide a valid  
metadata file.' );  
    $this->mo_saml_show_error_message();  
}  
if(!empty($_POST['metadata_url']))  
{  
    update_option( 'mo_saml_message', 'Please provide a valid  
metadata URL.' );  
    $this->mo_saml_show_error_message();  
}  
}  
  
}  
  
function get_current_customer() {  
    $customer = new CustomerSaml();  
    $content = $customer->get_customer_key();  
    if(!is_null($content)){  
        $customerKey = json_decode( $content, true );  
  
        $response = array();  
        if ( json_last_error() == JSON_ERROR_NONE ) {  
            update_option( 'mo_saml_admin_customer_key',  
$customerKey['id'] );  
            update_option( 'mo_saml_admin_api_key',  
$customerKey['apiKey'] );  
            update_option( 'mo_saml_customer_token', $customerKey['token']  
);  
            update_option( 'mo_saml_admin_password', " );  
            $certificate = get_option( 'saml_x509_certificate' );  
            if ( empty( $certificate ) ) {  
                update_option( 'mo_saml_free_version', 1 );  
            }  
        }  
    }  
}
```

```
        delete_option( 'mo_saml_verify_customer' );
        delete_option( 'mo_saml_new_registration' );
        $response['status'] = "success";
        return $response;
    } else {
        update_option( 'mo_saml_message', 'You already have an account
with miniOrange. Please enter a valid password.' );
        $this->mo_saml_show_error_message();
        //update_option( 'mo_saml_verify_customer', 'true' );
        //delete_option( 'mo_saml_new_registration' );
        $response['status'] = "error";
        return $response;
    }
}

function create_customer() {
    $customer = new CustomerSaml();
    $customerKey = json_decode( $customer->create_customer(), true );
    if(!is_null($customerKey)){
        $response = array();
        //print_r($customerKey);
        if ( strcasecmp( $customerKey['status'],
'CUSTOMER_USERNAME_ALREADY_EXISTS' ) == 0 ) {
            $api_response = $this->get_current_customer();
            //print_r($api_response);exit;
            if($api_response){
                $response['status'] = "success";
            }
            else
                $response['status'] = "error";
        }
    }
}
```

```

} else if ( strcasecmp( $customerKey['status'], 'SUCCESS' ) == 0 ) {
    update_option( 'mo_saml_admin_customer_key',
$customerKey['id'] );
    update_option( 'mo_saml_admin_api_key',
$customerKey['apiKey'] );
    update_option( 'mo_saml_customer_token', $customerKey['token']
);
    update_option( 'mo_saml_free_version', 1 );
    update_option( 'mo_saml_admin_password', " );
    update_option( 'mo_saml_message', 'Thank you for registering
with miniorange.' );
    update_option( 'mo_saml_registration_status', " );
    delete_option( 'mo_saml_verify_customer' );
    delete_option( 'mo_saml_new_registration' );
    $response['status']="success";
    return $response;
}

update_option( 'mo_saml_admin_password', " );
return $response;
}

function miniorange_sso_menu() {
    //Add miniOrange SAML SSO
    $slug = 'mo_saml_settings';
    add_menu_page( 'MO SAML Settings' . ___.('Configure SAML Identity Provider
for SSO', 'mo_saml_settings'), 'miniOrange SAML 2.0 SSO', 'administrator', $slug, array(
        $this,
        'mo_login_widget_saml_options'
    ), plugin_dir_url( __FILE__ ) . 'images/miniorange.png' );
}

```

```
add_submenu_page( $slug      , 'miniOrange SAML 2.0 SSO', 'Plugin
Configuration', 'manage_options', 'mo_saml_settings'
, array( $this, 'mo_login_widget_saml_options'));
add_submenu_page( $slug      , 'miniOrange SAML 2.0 SSO', 'Premium
Plans', 'manage_options', 'mo_saml_licensing'
, array( $this, 'mo_login_widget_saml_options'));
}

function mo_saml_authenticate() {
$redirect_to = '';
if ( isset( $_REQUEST['redirect_to'] ) ) {
$redirect_to = htmlentities( $_REQUEST['redirect_to'] );
}
if ( is_user_logged_in() ) {
if ( ! empty( $redirect_to ) ) {
header( 'Location: ' . $redirect_to );
} else {
header( 'Location: ' . site_url() );
}
exit();
}

}

function handleXmlError( $errno, $errstr, $errfile, $errline ) {
if ( $errno == E_WARNING && ( substr_count( $errstr,
"DOMDocument::loadXML()" ) > 0 ) ) {
return;
} else {
return false;
}
}
```

```
function mo_saml_plugin_action_links( $links ) {
    $links = array_merge( array(
        '<a href="' . esc_url( admin_url( 'admin.php?page=mo_saml_settings' ) ) . '">' .
        __( 'Settings', 'textdomain' ) . '</a>',
    ), $links );
    return $links;
}

function checkPasswordpattern($password){
    $pattern = '/^[(\w)*(!@\#\$\%^\&|^*\.\|-\\_)*]+$/';
    return !preg_match($pattern,$password);
}
new saml_mo_login;
```

Appendix N - GitHub w/ Wordpress Pages

As there are roughly 30+ files that are built out in the Wordpress page, we decided to consolidate this information by placing it all in a GitHub link that you can access to view this information and downloaded at your leisure. Wordpress pages and configuration files can all be viewed here: <https://github.com/COM-497/2020-Spring-Group4>

COM-497 / 2020-Spring-Group4

Watch 0 Star 1 Fork 0

Code Issues 0 Pull requests 2 Actions Projects 0 Wiki Security 0 Insights

No description, website, or topics provided.

- 35 commits	3 branches	0 packages	0 releases	3 contributors
Branch: master ▾	New pull request	Create new file Upload files Find file Clone or download ▾		
jdewinne856 WordPress files added			Latest commit f03f927 4 minutes ago	
HTML Max	Feb24 Max			2 months ago
MaxNode by Jared	April1			23 days ago
RAAAnodejs	CleanApril8			16 days ago
Wordpress Files	WordPress files added			4 minutes ago
visualize	Max Feb24.3			2 months ago
.DS_Store	WordPress files added			4 minutes ago
Calendar.tar.gz	CleanApril8			16 days ago

References

- Accime, E., (2019)., *Submit a form and receive email - Express Nodejs Nodemailer MailGun.* Retrieved from <https://youtu.be/JpcLd5UrDOQ>
- Baxter, J. (n.d.)., jebax - Overview. Retrieved from <https://github.com/jebax/SchejFrontend>
- ChChloeverity - Overview., (n.d.)., Retrieved from <https://github.com/chloeverity/SchejBackend>
- Jqense., (2020)., jqense/react-big-calendar. Retrieved from <https://github.com/intljusticemission/react-big-calendar>
- Saint Leo RA Resource Site., (n.d.)., Retrieved from <http://ras.saintleoresidencelife.com/>
- Templated Inc., (2016)., Visualize: A simple, one-page portfolio design with a fully functional lightbox., Retrieved from <https://templated.co/visualize>
- Traversy Media, (2017)., *Nodemailer - Send Emails From Your Node.js App.* (2017). Retrieved from <https://youtu.be/nF9g1825mwk>