GOMC Capstone

Product Design Specification

Version 1.0

November 1, 2017

Prepared by: Caleb Latimer

Ahmed Taher

Muamer Besic

VERSION HISTORY

Version #	Implemented By	Revision Date	Reason
0.1	Muamer Besic	10/15/17	Initial Design Definition draft
0.2	Muamer Besic	10/25/17	Added use case tables, security maintenance (hashing, salting)
0.3	Ahmed Taher	10/26/17	Added use case for downloads page.
0.4	Muamer Besic	10/26/17	Added sequence diagram, use case forgot password
0.5	Muamer Besic	10/27/17	Added use cases
0.5.5	Ahmed Taher	10/27/17	Added diagrams
0.6	Caleb Latimer	10/29/17	Added UI Design Modified Existing Use Cases Added in document hyperlinks Added Use Cases
0.7	Muamer Besic	10/30/17	Finalized document
0.8	Ahmed Taher & Muamer Besic	10/31/17	Fixes based on TA Feedback
0.9	Caleb Latimer	10/31/17	Modified Use Cases Added Use Case Diagram Added New Use Cases Added Communication Diagram Added Announcements Sequence Diagram
1.0	Muamer Besic & Ahmed Taher	11/1/17	Finalize, Convert, Review

Table of Contents

1. Introduction

1.1 Purpose of The Product Design Specification Document

The Product Design Specification document contains and tracks the necessary information required to effectively define architecture and system design in order to give the development team guidance on architecture of the system to be developed. The Product Design Specification document is created during the Planning Phase of the project. Its intended audience is the project manager, project team, and development team. Some portions of this document such as the user interface (UI) may on occasion be shared with the client/user, and other stakeholder whose input/approval into the UI is needed.

2. General Overview and Design Guidelines/Approach

This section describes the principles and strategies to be used as guidelines when designing and implementing the system. The most fundamental goal of this project is to develop a dynamic modern website as a medium core of presentation for the research, development, and new releases parts of the GOMC Project. The GPU Optimized Monte Carlo Method simulation engine is used to generate different classifications of molecular systems and using the Monte Carlo method, it aims to optimize the system by making different changes to it.

With the process of this simulation, the structure is eventually optimized to sustainable VLE. Once the engine reaches this point, the user is returned valuable data which can be used to optimize several factors about a chemical’s development, production, and storage. This project will encompass the full-scale design and implementation of a public website with the purpose of presenting a public face to the research and development of this engine by the Wayne State GOMC research group.

2.1 Assumptions / Constraints

It is assumed that the user has access to a modern web browser capable of rendering HTML5 and JavaScript. The modern web browsers that have this capability are Microsoft Edge version 40.15063.674.0, Google Chrome version 62.0.3202, and Firefox version 56.0.2. The website itself uses modern web development technologies which take advantage of HTML5. As such, without a browser the content cannot be viewed by the user and without a modern browser the content may not be displayed correctly.

Users are expected to view the website with a desktop machine running on either Windows 10 or a Linux distribution. The website will be viewable on MacOS but the software may not install or run properly.

When it comes to mobile use, users are also expected to view the website on mobile either with Android 7.0 or iOS 11. Tablet views are not supported, legacy browsers are not supported, and other mediums of web browsing such as screen-readers are also not supported.

Lastly, internet connection must be configured on the user’s device if it is a workstation or laptop. If on a mobile device, the user is expected to have access to at least 3G network capabilities and a mobile data plan in order to access the internet.

3. Architecture Design

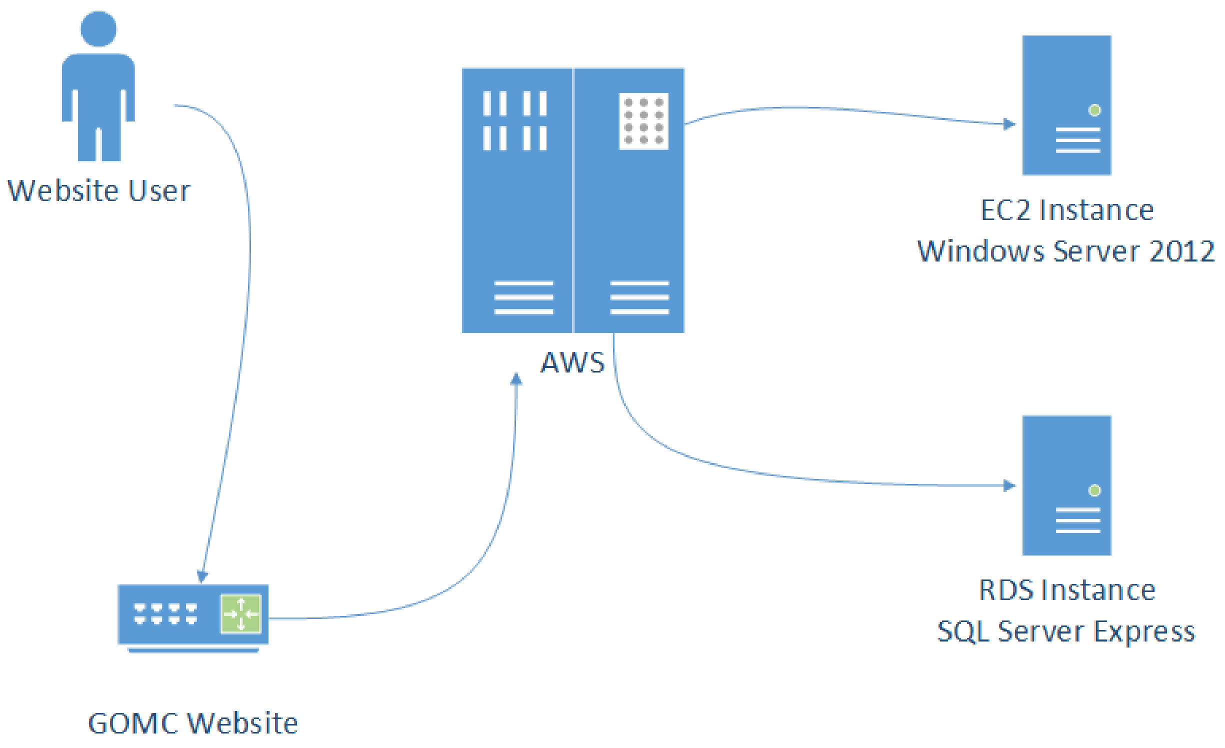
This section outlines the system and hardware architecture design of the system that is being built.

3.1 Hardware Architecture

The website is hosted using an AWS EC2 instance. The instance itself is a Windows Server 2012 while the database on the other hand is SQL Server Express.

Both instances are of type t2 micro. For a full description of t2 micro please visit the page https://aws.amazon.com/ec2/instance-types.

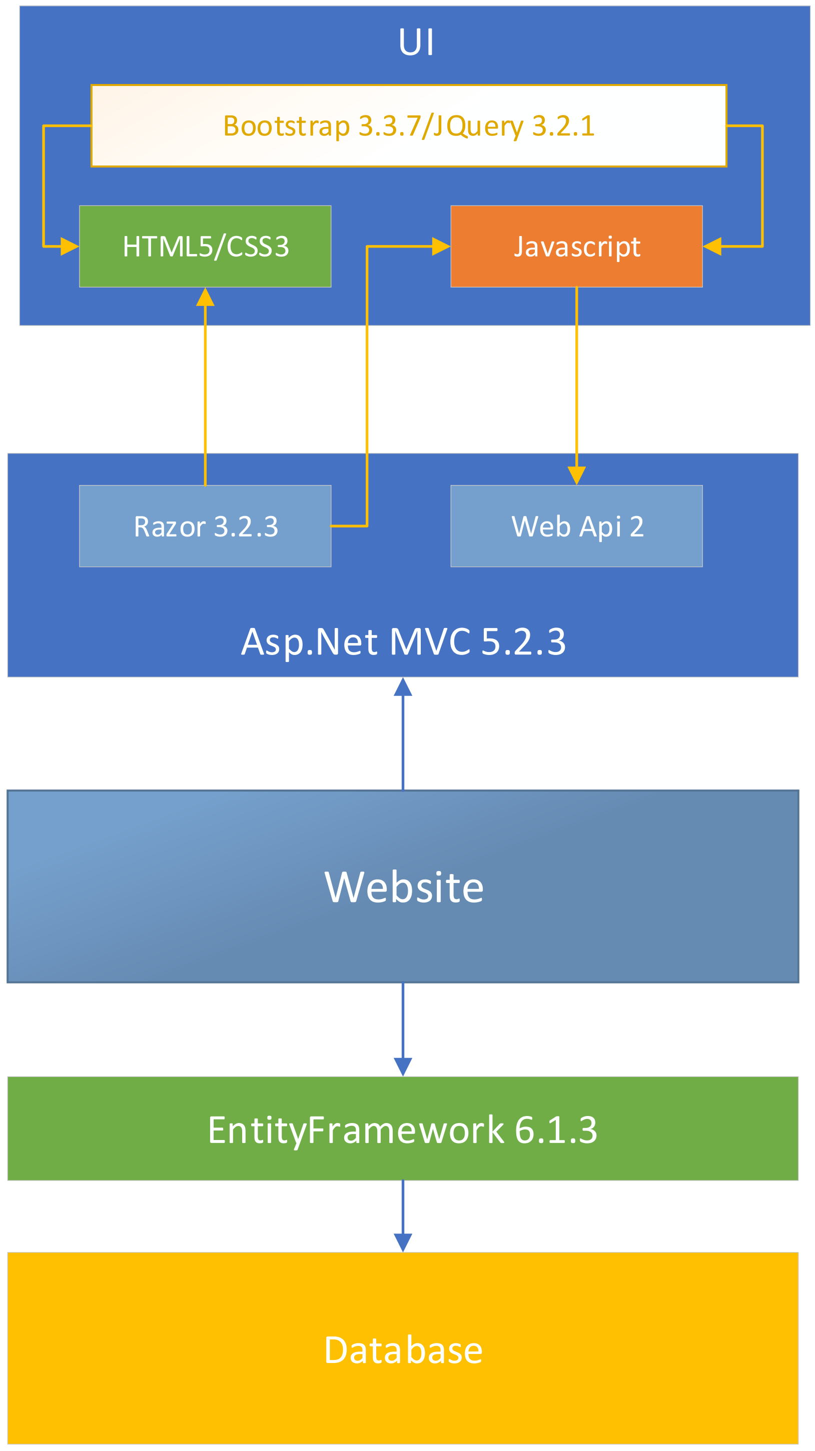
In short, the EC2 instance has 1GB of ram available and 1 vCPU on an Intel Xeon processor and 30GB of SSD storage. The RDS instance has 20GB of storage.



3.2 Software Architecture

The website will have a ASP.NET MVC 5.2.3 architecture and the choice for the view engine will be Razor. With the choice of Razor, we mean that when a user, for example, clicks on the Downloads page, Razor will render that page using the downloads Razor view which is part of the available source code. In addition, the website will utilize ASP.NET Web API for RESTful calls.

An example of RESTful calls the web app makes is form validation. After a form submit button is clicked, JavaScript code will handle and create AJAX requests, and specifying our Web API Controllers as the endpoints.


UI
Bootstrap 3.3.7/JQuery 3.2.1
HTML5/CSS3
Javascript
Razor 3.2.3
Web Api 2
Asp.Net MVC 5.2.3
Website
EntityFramework 6.1.3
Database

3.3 Security Architecture

Password Hashing

The GOMC website never transfers admin passwords in plain text. When it does transfer them, it makes sure the passwords are hashed when they are sent from the website to the database to check for matching hashing values. Password hashing works by taking a string of characters which in our case is the admin's password and creating a cryptic, constant sized password. With the hashing method, the GOMC website is the only component that handles admin passwords in plain text, and the GOMC database only stores passwords that have gone through the hashing process.

Salting

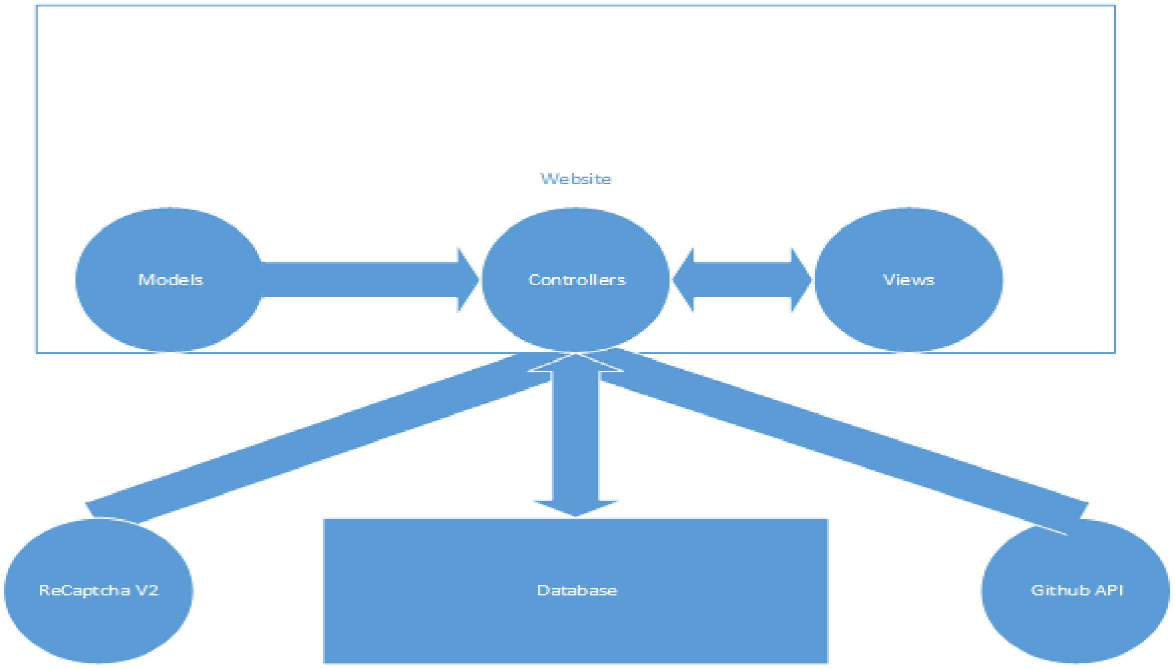
The GOMC website will be using hashing as well as salting when it comes to different layers of protection for passwords. Hashing is a good form of password protection however, it is also a common way of protecting a password so it does have well known workarounds where the most common being the dictionary attack. Libraries of common hash values and passwords that have been used by others or uncovered in successful attacks can be used. These hash values can be compared and the attacker will be able to guess what your hashing method might be and from there can also determine what the password might be.

Adding salting will add an appended value to the end of the password string inputted and once the salt value is added after the password, the password and salt value will become one string that is inputted for the entire password.

The salting value we are using is very long and it makes it very tough for a hacker to attempt to gain access to the admin account by figuring out the password hash.

3.4 Communication Architecture

Our communication architecture performs as a conventional dynamic website with some parts static rendered on a web server and other parts dynamic rendered through Razor from the ASP.NET framework as described in the previous section. In addition to that, our content delivery is dependent on external API's which are documented further in this document. Our approach will conform to standard HTTP protocols for client server communication.



4. System Design

4.1 Use-Cases

Use Case ID:	UC-1		
Use Case Name:	*Login*		
Created By:	*Muamer Besic*	***Last Updated By:***	Caleb Latimer

Date Created:	*10/25/17*	**Last Revision Date:**	*10/29/2017*
Actors:	*Webmaster or System Administrators*		
Description:	*This use case will describe the admin logging on to the admin page*		
Trigger:	*The admin wants to gain access to the admin page and admin features*		
Preconditions:	*The admin is not already logged in*		
Postconditions:	*Admin credentials will be authenticated and the admin will be redirected to the admin view*		
Normal Flow:	*1. System validates the admin's email/password and logs them in* *2. System displays the admin page*		
Alternative Flows:	*N/A*		
Exceptions:	*1. Invalid email/password* *If incorrect information is entered, an error message will appear specifying which credentials are invalid* *2. User can retype information or choose not to do anything, at which point this use case ends*		
Frequency of Use:	*Whenever the admin wants to access the admin page*		
Assumptions:	*The admin wants to get on the admin page and has navigated themselves to the /admin page to log in*		

Use Case ID:	UC-2		
Use Case Name:	*Logout*		
Created By:	*Muamer Besic*	**Last Updated By:**	Caleb Latimer
Date Created:	*10/25/17*	**Last Revision Date:**	*10/29/2017*
Actors:	*Webmaster or System Administrators*		
Description:	*This use case will describe the admin logging out of admin page*		

Trigger:	*The admin wants to logout of admin page once they have finished using it*
Preconditions:	*1. Admin is logged in* *2. Admin wants to log out*
Postconditions:	*Admin credentials will be logged out and the admin will be redirected to the login page as a general user*
Normal Flow:	*1. Admin is done using the admin page* *2. Admin clicks on logout button* *3. System logs the admin out and redirects them to the login page*
Alternative Flows:	*N/A*
Exceptions:	*N/A*
Frequency of Use:	*Whenever an admin is logged in*
Assumptions:	*The user is expected to have a system that conforms to the constraints specified in section 2.1* *The server and backend must be operational*

Use Case ID:	UC-3		
Use Case Name:	*Downloads & Examples Links*		
Created By:	*Ahmed Taher*	**Last Updated By:**	*Caleb Latimer*
Date Created:	*10/26/17*	**Last Revision Date:**	*10/29/17*
Actors:	*Website User*		
Description:	*This use case will describe a user who has accessed the Downloads Page on the website*		
Trigger:	*The user requests the downloads page.*		
Preconditions:	*The user has access to either the website navigation bar or the URL for the downloads page.*		
Postconditions:	*The list of latest files for the GOMC application will be displayed as*		

	links.
Normal Flow:	1. *The user clicks on the link titled "Downloads" in the website navbar.* 2. *The back-end pulls the list of latest files from the Web API.* 3. *The user is shown links to the files.*
Alternative Flows:	1. *The user manually enters the URL of the downloads page into the browser URL bar.* 2. *The back-end pulls the list of latest files from the Web API.* 3. *The user is shown links to the files.*
Exceptions:	*Same as UC-13*
Frequency of Use:	*Whenever the user wants the latest files of the GOMC application.*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-4		
Use Case Name:	*Registration Form*		
Created By:	*Muamer Besic*	***Last Updated By:***	Caleb Latimer
Date Created:	*10/27/17*	***Last Revision Date:***	*10/29/2017*
Actors:	*Website User*		
Description:	*This use case will describe a user registering on GOMC website*		
Trigger:	*The user requests the downloads page*		
Preconditions:	*User has loaded the downloads page*		
Postconditions:	*User's email and password are stored in our database*		
Normal Flow:	1. *Form asks user to enter basic information* 2. *User fills in all required fields* 3. *User clicks to verify captcha* 4. *User clicks register*		

	5. *Front-end validation gets initiated and sends data to the back-end for additional validation* 6. *Validation from UC-8 returns successfully* 7. *System stores information and registers email* 8. *The form slides up and the input button is disabled with the message "Thank you for registering!"*
Alternative Flows:	*N/A*
Exceptions:	1. *Invalid information was entered such as invalid email format etc. This triggers a feedback from the front-end validation which appears on the UI for the user* 2. *User doesn't click on captcha to confirm they are not a bot which doesn't allow them to click register button as it will be disabled and an error message will display telling them to either fill out the form or to do the captcha*
Frequency of Use:	*Whenever a new user wants to register their email*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-5		
Use Case Name:	*Edit website*		
Created By:	*Muamer Besic*	***Last Updated By:***	Caleb Latimer
Date Created:	*10/27/17*	***Last Revision Date:***	*10/29/2017*
Actors:	*Webmaster or System Administrators*		
Description:	*This use case will describe admin editing content on website*		
Trigger:	*Admin interacts with the modification controls that are part of the admin view*		
Preconditions:	*Admin is logged in*		
Postconditions:	*Admin can edit information on GOMC website*		
Normal Flow:	1. *Admin logs in* 2. *Admin can post new announcements, manually refresh the*		

	downloads list, manually refresh the examples list and upload new documentation
Alternative Flows:	*N/A*
Exceptions:	*1. Admin is not able to login because they forgot their login information* *2. Admin loses internet connection while logged in and is not able to save changes made*
Frequency of Use:	*Whenever an admin wants to make changes*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-6		
Use Case Name:	*XML Input page*		
Created By:	*Muamer Besic*	***Last Updated By:***	Caleb Latimer
Date Created:	*10/27/17*	***Last Revision Date:***	*10/29/17*
Actors:	*Website User*		
Description:	*This use case will describe the process of a user inputting data into the XML input page and getting an XML form back*		
Trigger:	*User clicks the "Start Form" button on the downloads page and is redirected to the XML Input page*		
Preconditions:	*Input fields on input page are blank*		
Postconditions:	*Input field values are fully completed and are returned to the user as a well-formed XML download*		
Normal Flow:	*1. User goes to downloads page with the form-launch button* *2. User clicks button and reaches the XML Input page* *3. User inputs data into fields and uses the arrow buttons to navigate through the form* *4. User's choice either disable or remove certain options that are no longer valid on further pages in the form* *5. User clicks "Create XML input" after filling out the form and sends data to the backend of the*		

	6. *Download of the XML file is sent back to the user's machine*
Alternative Flows:	*1. User inputs invalid information which causes an error message to pop up next to the invalid field saying, "Input is invalid"* *2. User forgets to input data in a field which in return will not allow the user to click and move on to the next panel until the user inputs all needed data*
Exceptions:	1. *User is not sure of what to input in each field* 2. *User inputs invalid information* 3. *User skips a field they were supposed to fill out or bubble they were supposed to click on which doesn't allow them to proceed to the next part of the form* 4. *If the user makes a mistake on the last page of the form they will not be allowed to create the XML*
Frequency of Use:	*Each time a user wants an XML form created*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-7		
Use Case Name:	*XML Downloadable Configuration File*		
Created By:	*Muamer Besic*	***Last Updated By:***	Caleb Latimer
Date Created:	*10/27/17*	***Last Revision Date:***	*10/29/17*
Actors:	*Website User*		
Description:	*This use case will go over how XML Schema outputs files*		
Trigger:	*User has clicked on Create XML Form*		
Preconditions:	*User has correctly filled out input page form*		
Postconditions:	*User will receive output files for XML schema*		
Normal Flow:	1. *Create XML Form button is clicked* 2. *Input fields are sent to api/configInput/FormPost* 3. *Form data is wrapped into a dictionary of key-value pairs* 4. *A model is made from the resulting dictionary which will be mapped to a guid and returned as a downloadable link*		

Alternative Flows:	*N/A*
Exceptions:	1. *User doesn't fill out input page correctly* 2. *User gets an error message as the dictionary to model conversion will throw an error* 3. *Error will be sent back to the UI and display meaningful response as an alert box*
Frequency of Use:	*Each time user is filling out input page form*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-8		
Use Case Name:	*Registration Form- Captcha Processing*		
Created By:	*Caleb Latimer*	***Last Updated By:***	
Date Created:	10/29/17	***Last Revision Date:***	
Actors:	*Website Users*		
Description:	*This use case describes the actions of the reCAPTCHA V2 that we have implemented on the registration form*		
Trigger:	*User clicks the captcha box to verify that they are not a bot*		
Preconditions:	*The submit button is disabled with a message that displays the text "Please fully fill out the form in order to submit"*		
Postconditions:	*The user is verified as not being a bot and the submission button becomes clickable*		
Normal Flow:	1. *The user clicks the reCAPTCHA area* 2. *The user completes the captcha successfully* 3. *A post is sent to the /api/Registration/Input endpoint from the front-end* 4. *The Registration Controller must check for the captcha being valid by sending the secret-key, response of the captcha and the user ip to Google's endpoint and return the result* 5. *Front end waits for the result of a post from the backend to the Google and handles it by the happy-path of hiding the form and disabling the close form button and modifying it read "Thank you*		

	for registering"
Alternative Flows:	*N/A*
Exceptions:	1. *Captcha returns a negative result or the user didn't fill out the captcha* 2. *Submit button remains grayed out, and a validation message provides feedback that the user missed filling it out or that they need to fill out again*
Frequency of Use:	*For every user that needs to register for emails from GOMC*
Assumptions:	*Same as UC-2* *The endpoint from the google api must be up and running to authenticate*

Use Case ID:	UC-9		
Use Case Name:	*Admin View*		
Created By:	*Caleb Latimer*	***Last Updated By:***	
Date Created:	10/29/17	***Last Revision Date:***	
Actors:	*System Administrators or Webmaster*		
Description:	*This use case will describe the view returned to the Admin upon a successful login*		
Trigger:	*Admin logs into the site*		
Preconditions:	*Admin credentials are validated in accordance the UC-1*		
Postconditions:	*Admin only statistics are displayed*		
Normal Flow:	1. *Admin logs in and is validated* 2. *Admin can see the following information with respect to the site* a. *A table reflecting all registered users* b. *A form with a text area to update the announcements section of the landing page* c. *Buttons to manually refresh the downloads, examples and latex document* d. *Section for presentation of specially configured table from*		

	the database in a read-only format
Alternative Flows:	*N/A*
Exceptions:	*1. Admin login fails* *2. Admin exception case executes as described in UC-1*
Frequency of Use:	*Applies every time the admin logs in*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-10		
Use Case Name:	*Latex Documentation*		
Created By:	*Caleb Latimer*	***Last Updated By:***	
Date Created:	10/29/17	***Last Revision Date:***	
Actors:	*Website User*		
Description:	*User opts to view the html or pdf version of the documentation*		
Trigger:	*User clicks either the html or pdf buttons on the downloads page*		
Preconditions:	*User is already on the downloads page and clicks either of the documentation buttons*		
Postconditions:	*User is provided with the documentation*		
Normal Flow:	*1. Use clicks the html button* *2. A redirect occurs taking the user to a child page that has the documentation uploaded by performing a GET request* *3. The page has the documentation and a search bar allowing the user to get to specific headers in the document faster*		
Alternative Flows:	*1. User clicks the pdf button* *2. A pdf is opened in another tab with the documentation in view by performing a GET request*		
Exceptions:	*Should any exceptions be thrown from the back-end an alert box prompts the user to refresh the page and try again*		

Frequency of Use:	*Any instance where the user will try to view the documentation*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-11		
Use Case Name:	*Website interaction*		
Created By:	*Caleb Latimer*	***Last Updated By:***	
Date Created:	*10/29/17*	***Last Revision Date:***	
Actors:	*Website User*		
Description:	*This use case describes the process of the user navigating to the main page of the website*		
Trigger:	*The user enters the website URL into a web browser*		
Preconditions:	*User has an open web browser*		
Postconditions:	*User navigates to the landing page of the website*		
Normal Flow:	*1. User opens web browser 2. User enters URL of gomc.eng.wayne.edu or ahtaher.net 3. The current website loads into the user's browser 4. The landing page is displayed*		
Alternative Flows:	*1. User opens web browser 2. User searches "Wayne State GOMC" or related query 3. A link to the gomc.eng.wayne.edu page appears in the top 5 results 4. Clicking the link brings the user to the landing page*		
Exceptions:	*N/A*		
Frequency of Use:	*Any time the user wants to navigate to the website*		
Assumptions:	*Same as UC-2*		

Use Case ID:	UC-12

Use Case Name:	_Navigation bar & Site Navigation - Normal_		
Created By:	_Caleb Latimer_	**_Last Updated By:_**	
Date Created:	_10/29/17_	**_Last Revision Date:_**	
Actors:	_Website User_		
Description:	_This use case describes User Navigation to different pages through the site_		
Trigger:	_User clicks a link in the navigation bar or footer_		
Preconditions:	_The user is already on the website_		
Postconditions:	_User is on a different page in the website_		
Normal Flow:	_1. User is on a page in the website 2. User clicks a navigation bar link 3. User is taken to a different page in the site 4. Navigation bar bottom border is highlighted showing the user position_		
Alternative Flows:	_1. User is on a page in the website 2. User clicks on a link in the footer link 3. User is taken to a different page in the site 1. User is on a page in the website 2. User types /linkName to navigate to a different page in the site 3. User is navigated to a different page, if it's a navbar page the border bottom is shown if it's a footer page then no visual indicator is shown_		
Exceptions:	_1. User improperly types a navigation link 2. An error is shown and the user can either re-type their entry or click the back button in their browser to get back to the previous page_		
Frequency of Use:	_Applies every time that the user navigates to a different page_		
Assumptions:	_Same as UC-2_		

Use Case ID:	UC-13

Use Case Name:	*Navigation bar- Mobile*		
Created By:	*Caleb Latimer*	**Last Updated By:**	
Date Created:	*10/29/17*	**Last Revision Date:**	
Actors:	*Website User*		
Description:	*This use case describes User Navigation to different pages through the site*		
Trigger:	*User taps the hamburger icon*		
Preconditions:	*User has navigated to main site via a mobile device*		
Postconditions:	*User can navigate to different parts of the site*		
Normal Flow:	*1. User navigates to website 2. User taps the hamburger icon 3. The navigation menu slides in from the right with the current choice highlighted 4. Tapping a different choice updates the highlight, slides the menu back in and loads the new page*		
Alternative Flows:	*1. User navigates to the website 2. User taps one of the links in the footer 3. Page loads and behaves the same as in UC-13*		
Exceptions:	*Same as UC-12*		
Frequency of Use:	*Same as UC-12*		
Assumptions:	*Same as UC-2*		

Use Case ID:	UC-15		
Use Case Name:	*Admin Page- Post New Announcements*		
Created By:	*Caleb Latimer*	**Last Updated By:**	
Date Created:	*10/31/17*	**Last Revision Date:**	

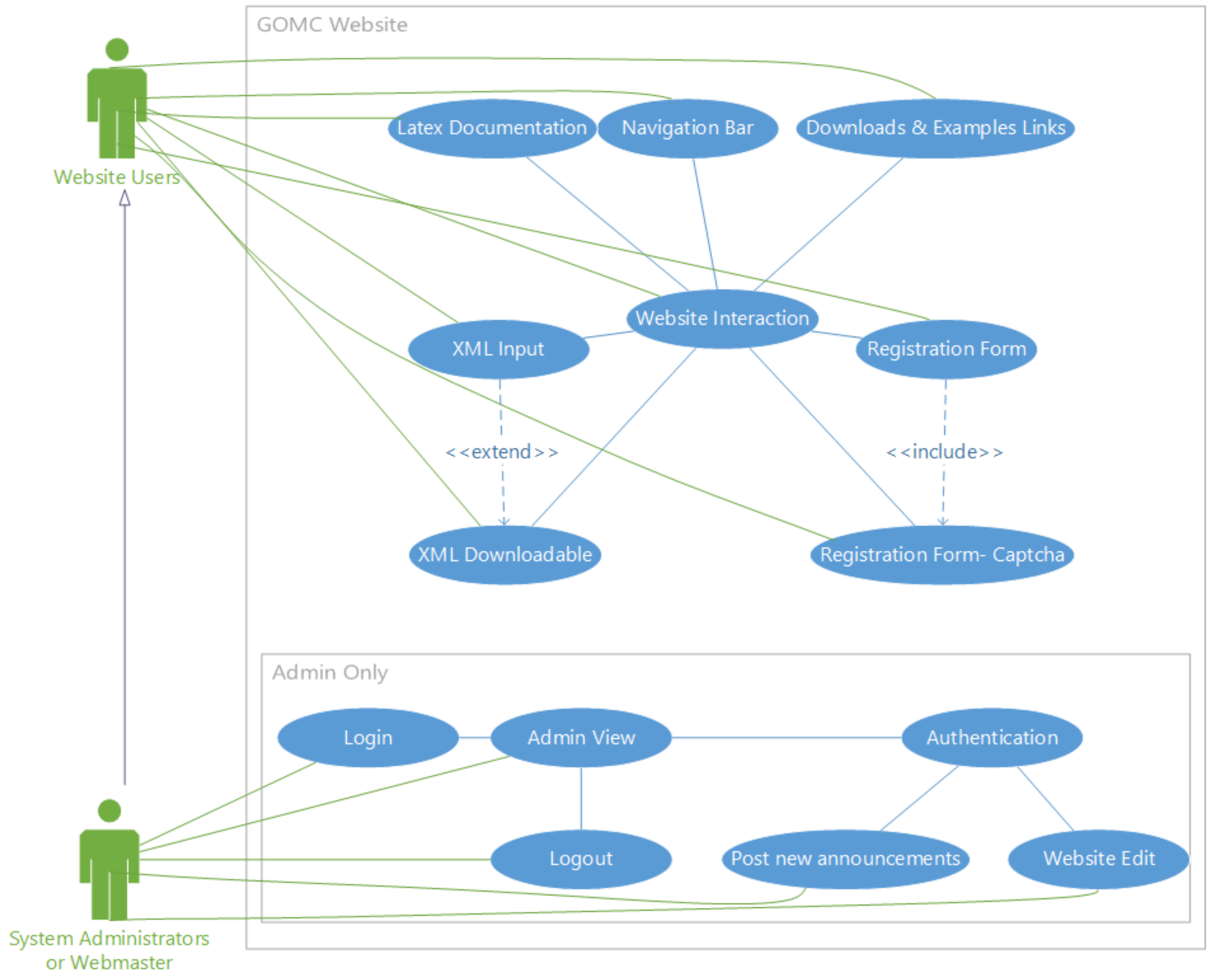
Actors:	*System Administrators or Webmaster*
Description:	*This Use Case will describe the event of the admin adding a new announcement*
Trigger:	*The Admin is logged into the admin view and desires to add a new announcement*
Preconditions:	*Website is fully functional* *Admin is logged in*
Postconditions:	*GOMC landing page has a new announcement prepended to the list*
Normal Flow:	*1. Admin clicks into the text-area to type a new announcement text* *2. Admin clicks the "Post" button* *3. Authentication from UC-16 runs* *4. UC-16 returns positive and message is posted to the announcements section of the landing page* *5. The text area has a message above it that reads "You have posted a new message click here to read" with the word "read" as a hyperlink* *6. Clicking to read the message navigates the admin to the landing page so that they can view the new announcement* *7. Upon returning to the admin page the message is gone*
Alternative Flows:	*1. Admin clicks into text-area to generate a new message text* *2. Admin clicks the "Reset" button* *3. Text-area is cleared* *4. Admin enters new text* *5. Admin clicks the "Post" button* *6. Authentication check from UC-16 runs* *7. UC-16 returns positive and message is posted to the announcements section of the landing page* *8. The text area has a message above it that reads "You have posted a new message click here to read"* *9. Admin refreshes the page* *10. Upon reload the message is gone*
Exceptions:	*1. Admin clicks into text-area to generate a new message text* *2. Admin clicks the "Post" button* *3. Authentication check from UC-16 runs* *4. UC-16 returns negative and message is deleted* *5. The exception case of UC-16 takes over this exception flow and executes fully*

Frequency of Use:	*Any occasion where the admin wants to add a new announcement*
Assumptions:	*Same as UC-2*

Use Case ID:	UC-16		
Use Case Name:	*Website Admin - Admin interaction authentication*		
Created By:	*Caleb Latimer*	***Last Updated By:***	
Date Created:	*10/31/17*	***Last Revision Date:***	
Actors:	*System Administrators or Webmaster*		
Description:	*This Use Case will describe the underlying process of checking the administrator's admin status before they perform each action*		
Trigger:	*The admin would like to update some website content*		
Preconditions:	*UC-15*		
Postconditions:	*The action that the admin takes is performed*		
Normal Flow:	*1. A loading glyph appears in place of text-area to indicate to the user that something is processing 2. The GUID of the successful login is checked against the backend to ensure a valid session is running 3. Check returns successfully and text is fully processed*		
Alternative Flows:	*N/A*		
Exceptions:	*1. A loading glyph appears in place of text-area to indicate to the user that something is processing 2. The GUID of the successful login is checked against the backend to ensure a valid session is running 3. Check returns unsuccessfully and text is halted 4. The running session is terminated 5. The admin is redirected to the admin login page with an alert box*		

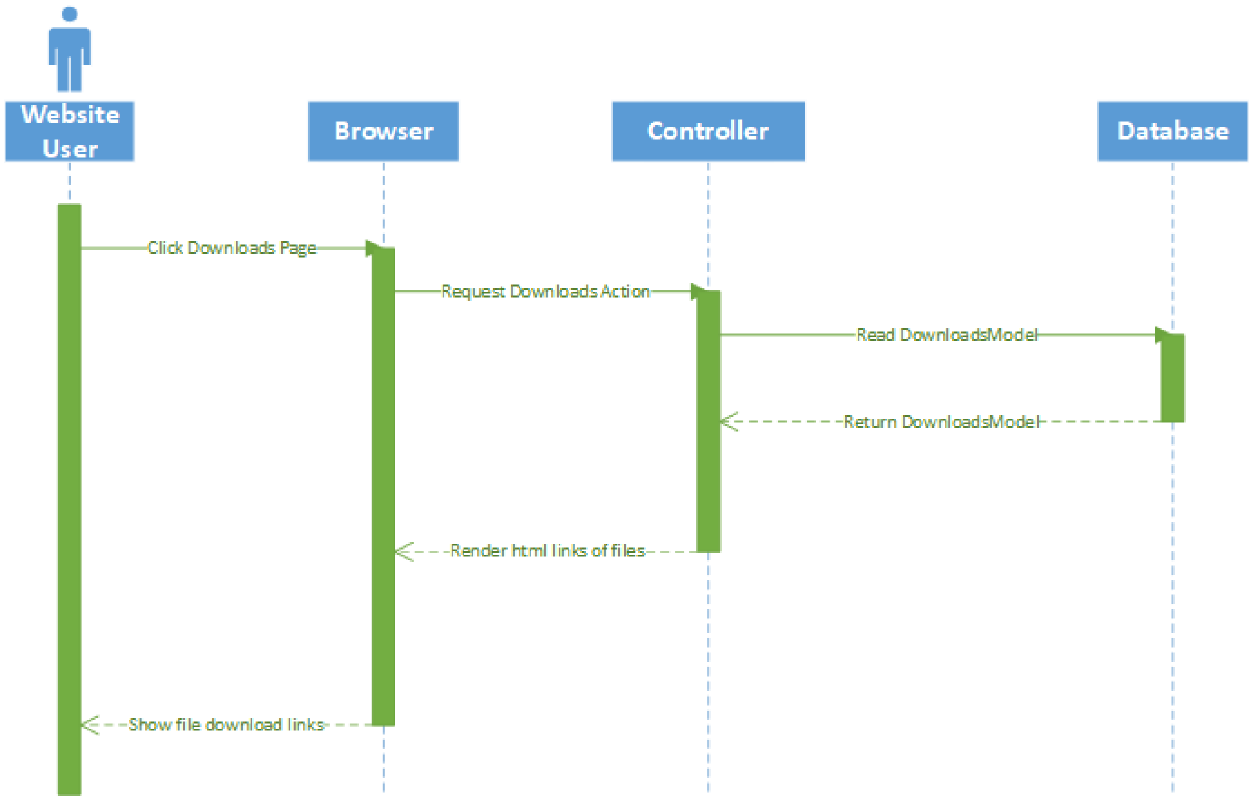
	that reads: "Your session has expired, please log in again" *6. Upon attempting to login UC-1 normal flow is initiated*
Frequency of Use:	*Any instance of the admin interacting with the dynamic content controls within the admin view*
Assumptions:	*Same as UC-2*

4.2 Use Case Diagram

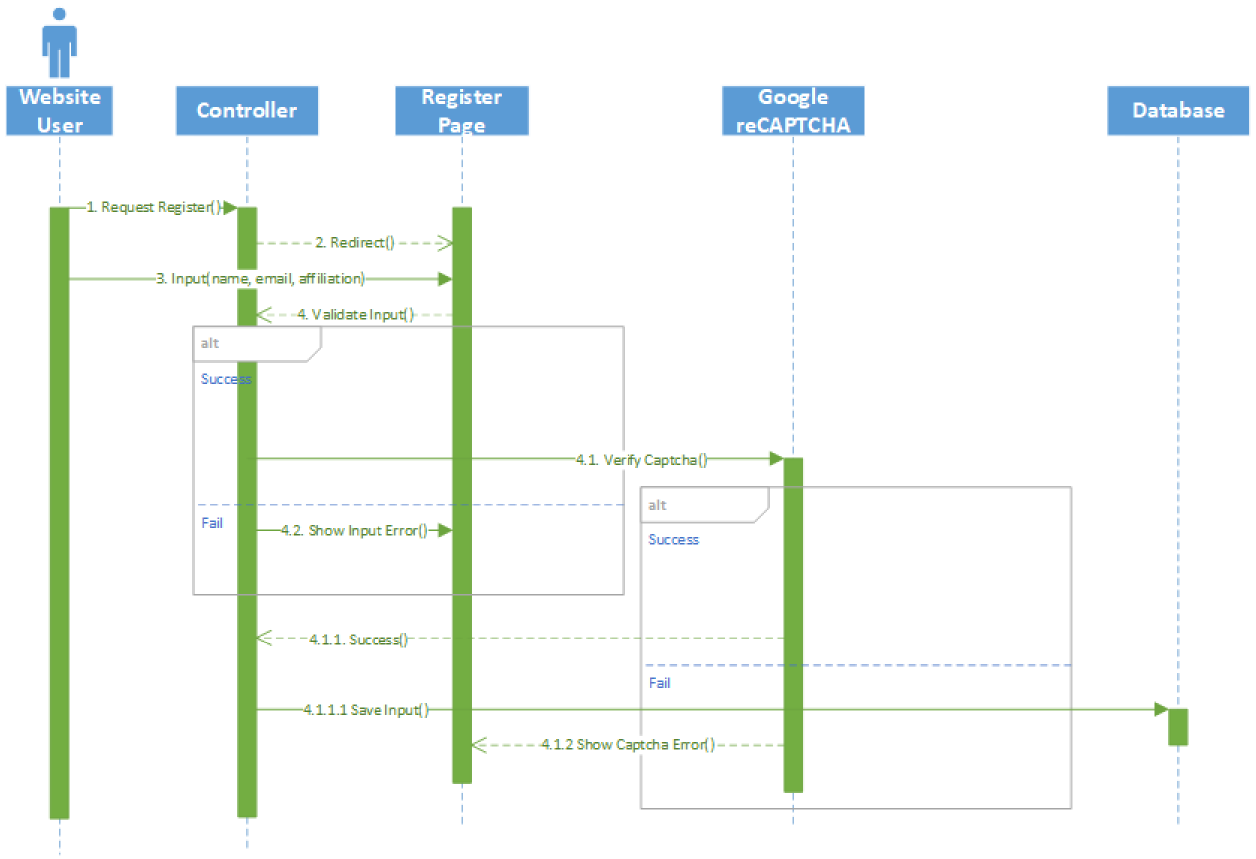


4.3 Sequence Diagrams

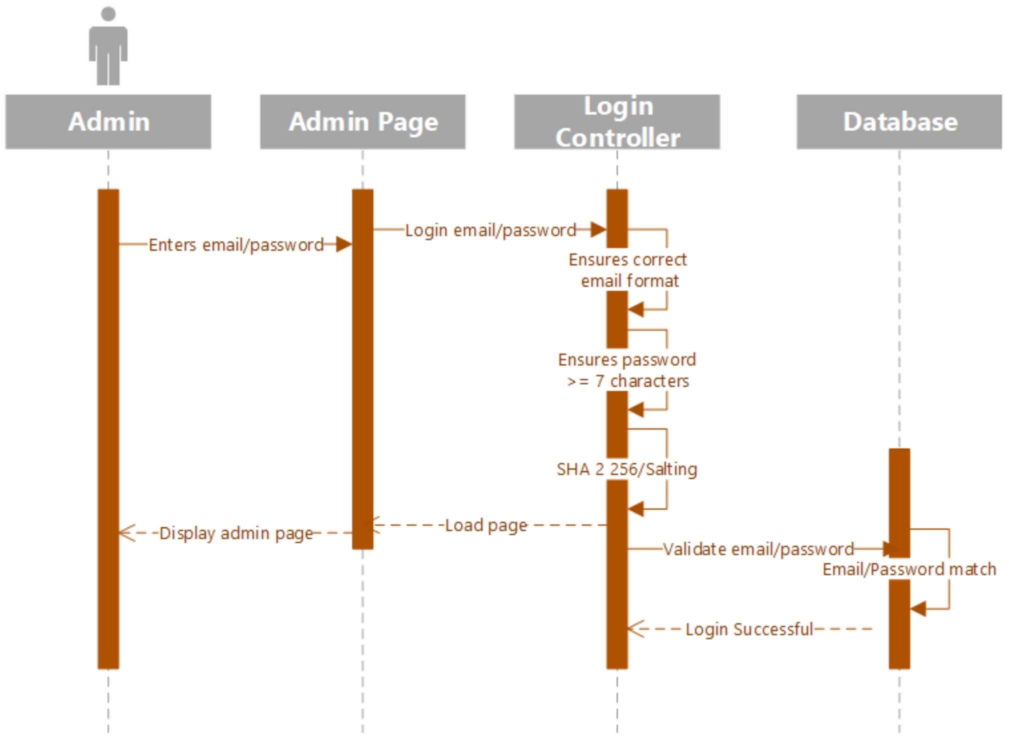
Downloads page



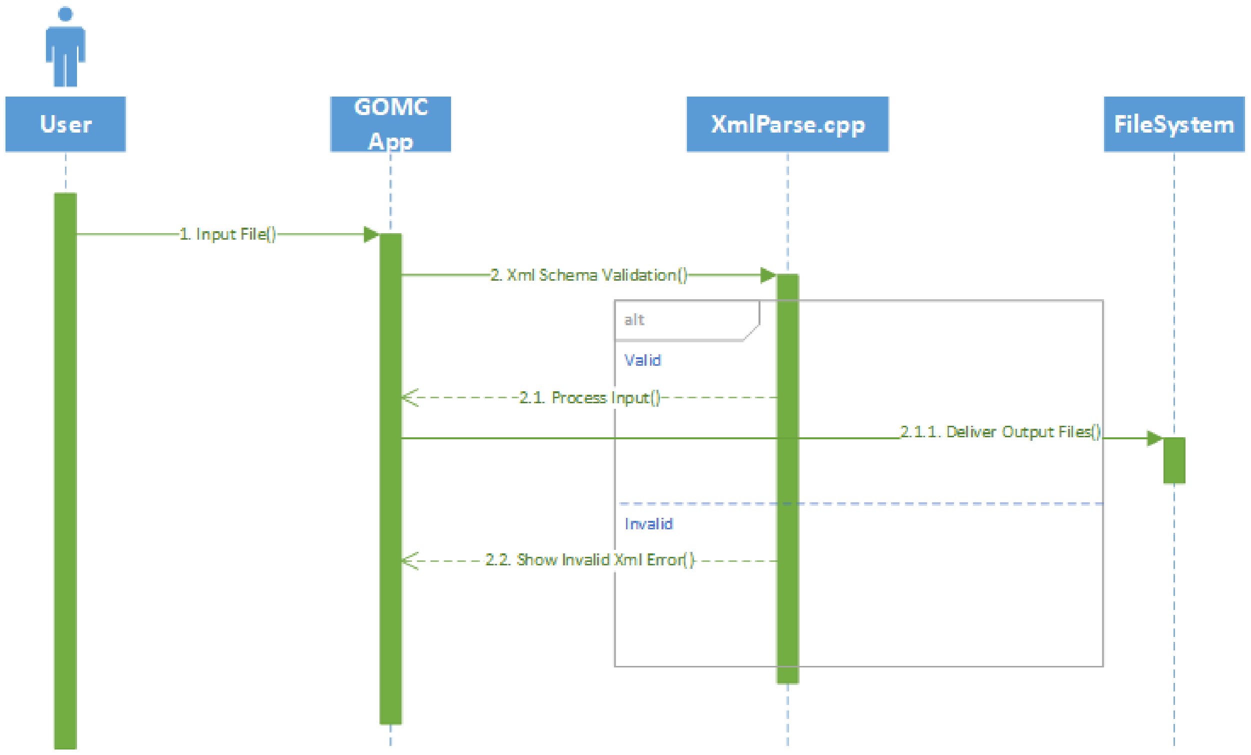
Registration



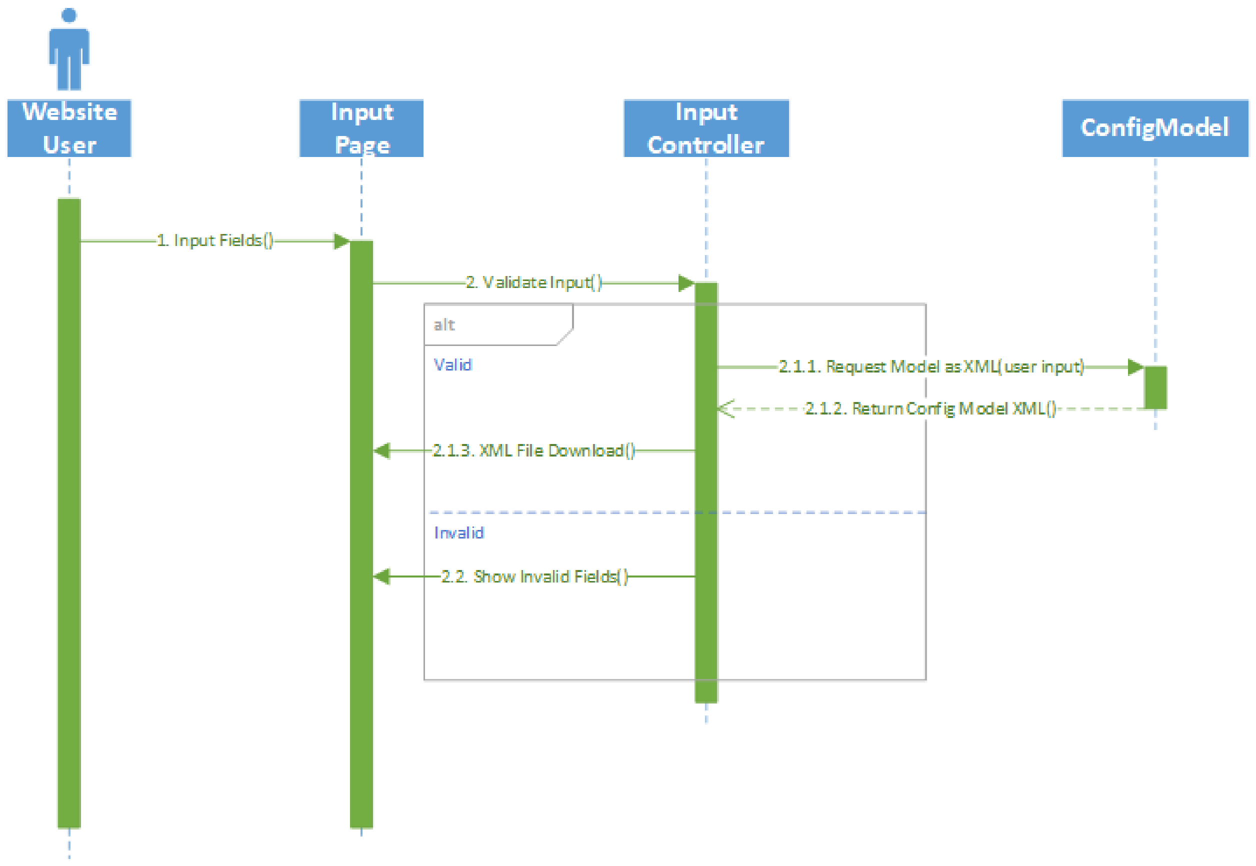
Admin Login


Admin
Admin Page
Login Controller
Database
Enters email/password
Login email/password
Ensures correct email format
Ensures password >= 7 characters
SHA 2 256/Salting
Load page
Display admin page
Validate email/password
Email/Password match
Login Successful

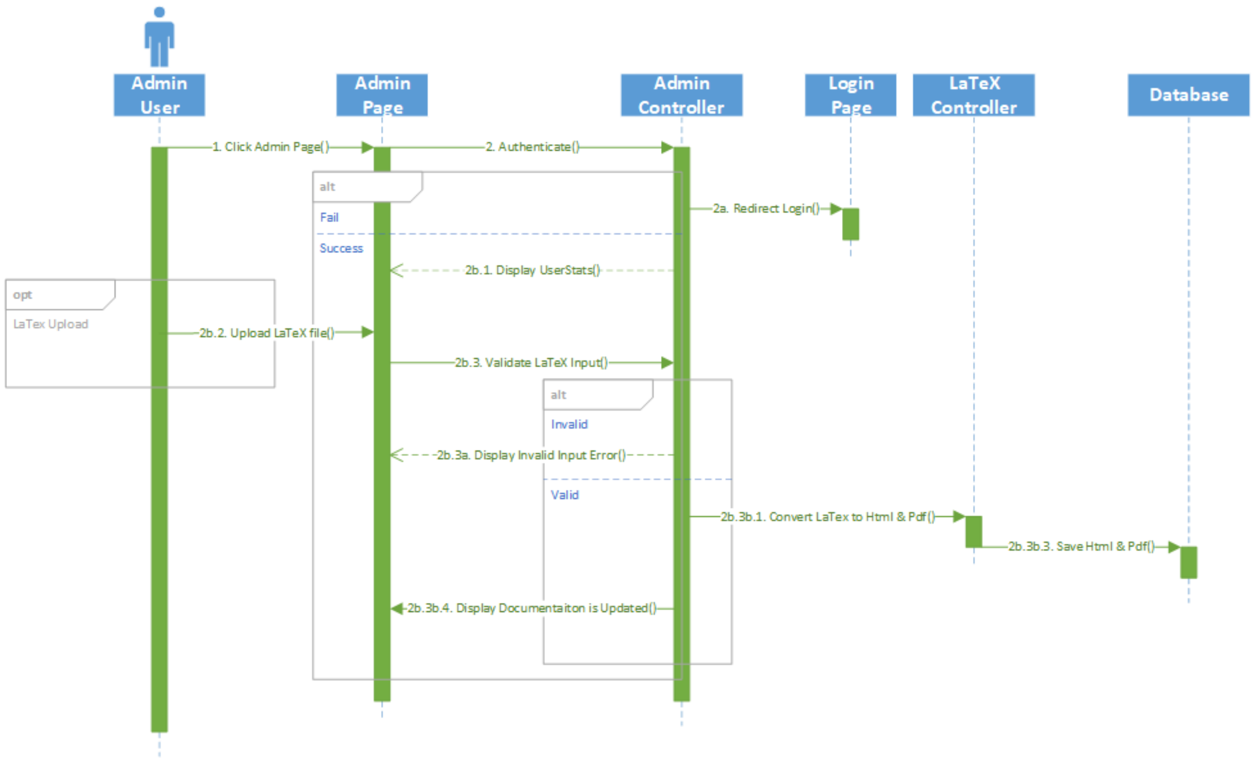
GOMC App XML Parser


User
GOMC App
XmlParse.cpp
FileSystem
1. Input File()
2. Xml Schema Validation()
alt
Valid
2.1. Process Input()
2.1.1. Deliver Output Files()
Invalid
2.2. Show Invalid Xml Error()

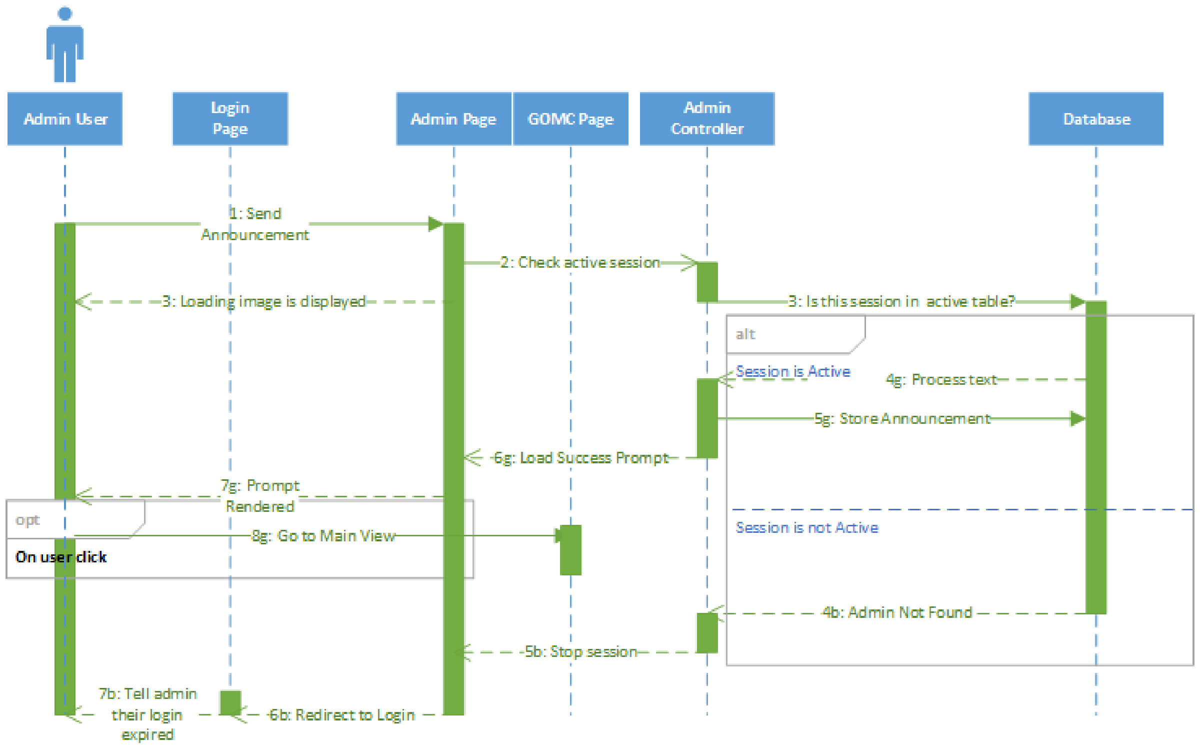
Input Page


Website User
Input Page
Input Controller
ConfigModel
1. Input Fields()
2. Validate Input()
alt
Valid
2.1.1. Request Model as XML(user input)
2.1.2. Return Config Model XML()
2.1.3. XML File Download()
Invalid
2.2. Show Invalid Fields()

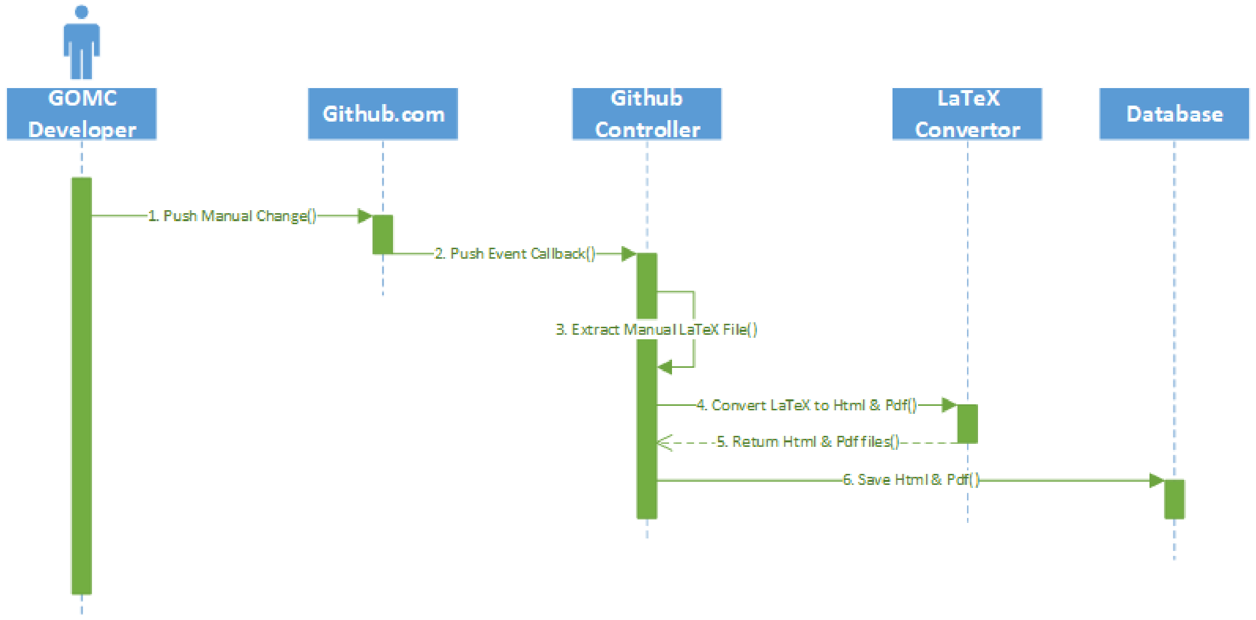
Admin Page


Admin User
Admin Page
Admin Controller
Login Page
LaTeX Controller
Database
1. Click Admin Page()
2. Authenticate()
alt
Fail
2a. Redirect Login()
Success
2b.1. Display UserStats()
opt
LaTex Upload
2b.2. Upload LaTeX file()
2b.3. Validate LaTeX Input()
alt
Invalid
2b.3a. Display Invalid Input Error()
Valid
2b.3b.1. Convert LaTex to Html & Pdf()
2b.3b.3. Save Html & Pdf()
2b.3b.4. Display Documentaiton is Updated()

Admin Announcements & Interaction Authentication

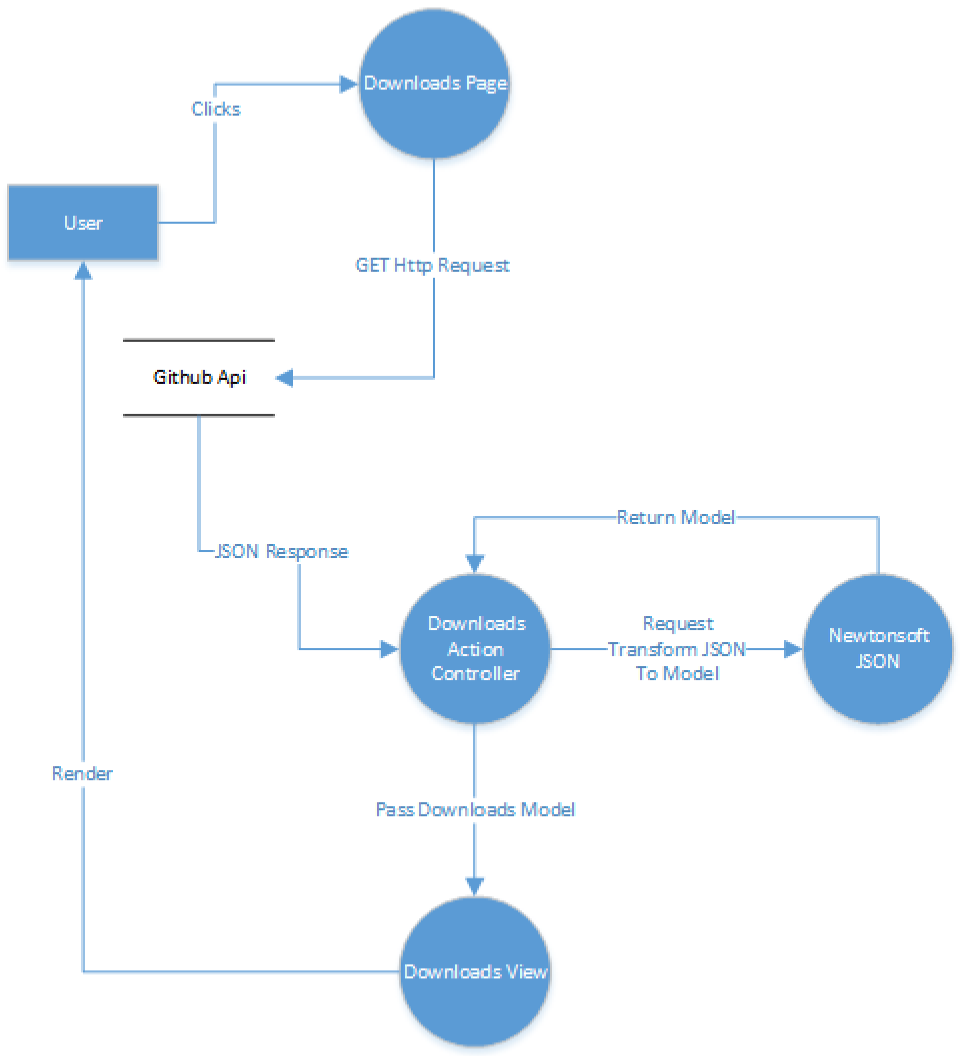


GitHub Push Event Web hook

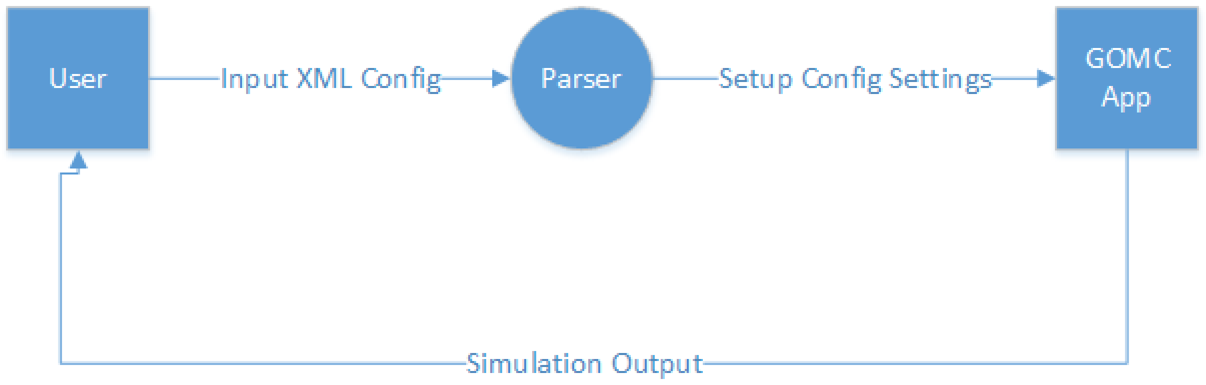


4.4 Data Flow Diagram

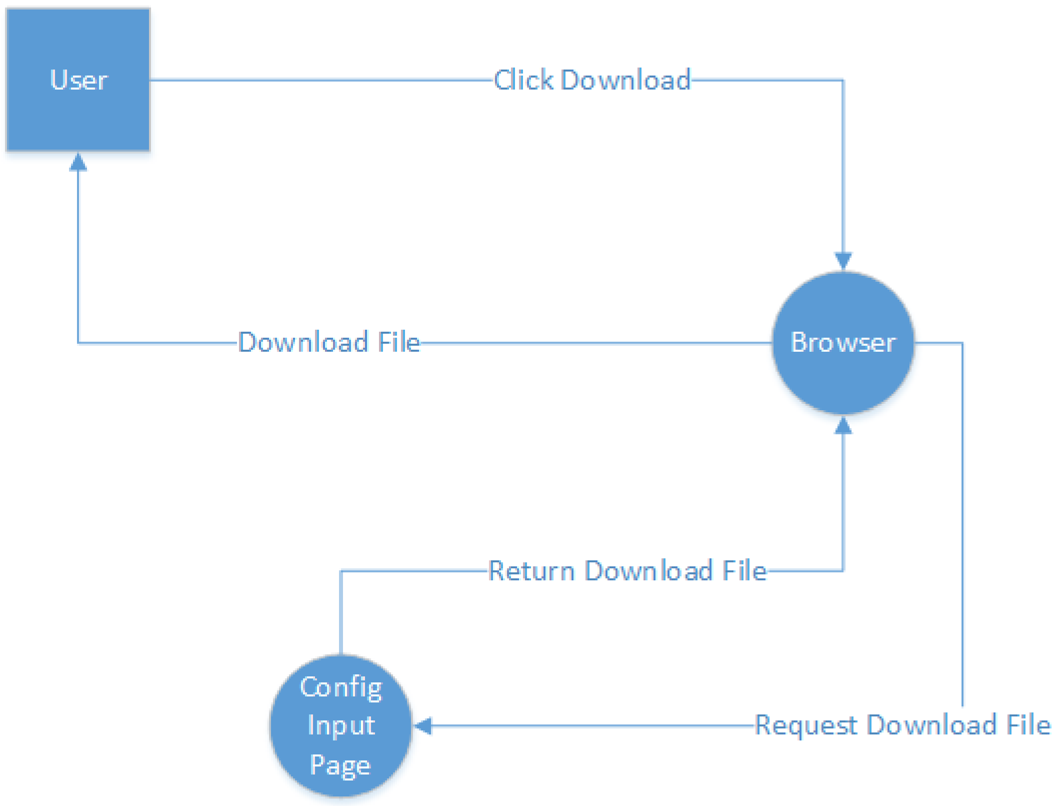
Downloads Page



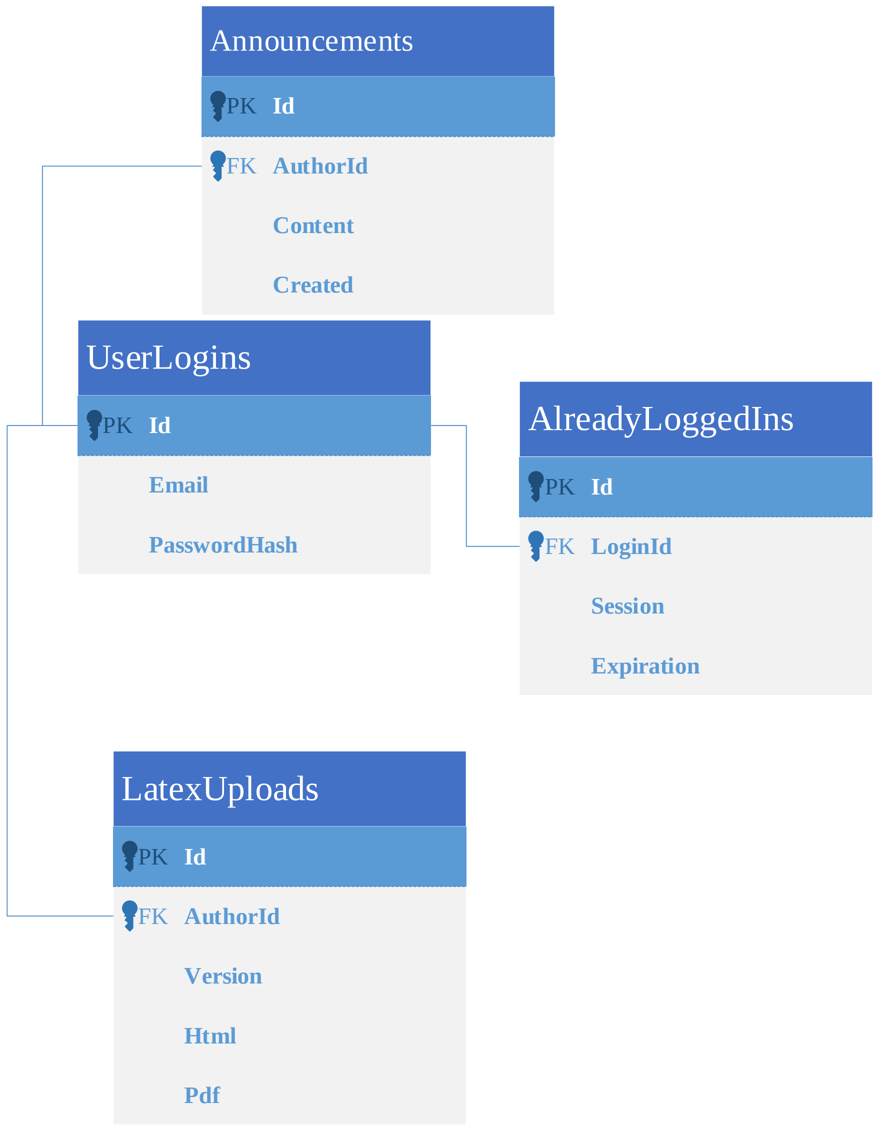
XML Parser

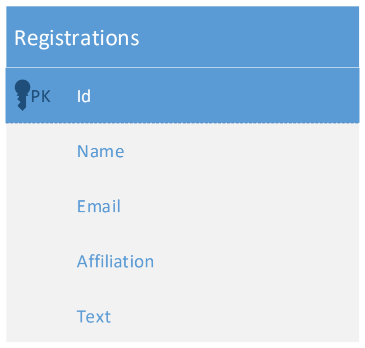


Config Page



4.5 Database Design



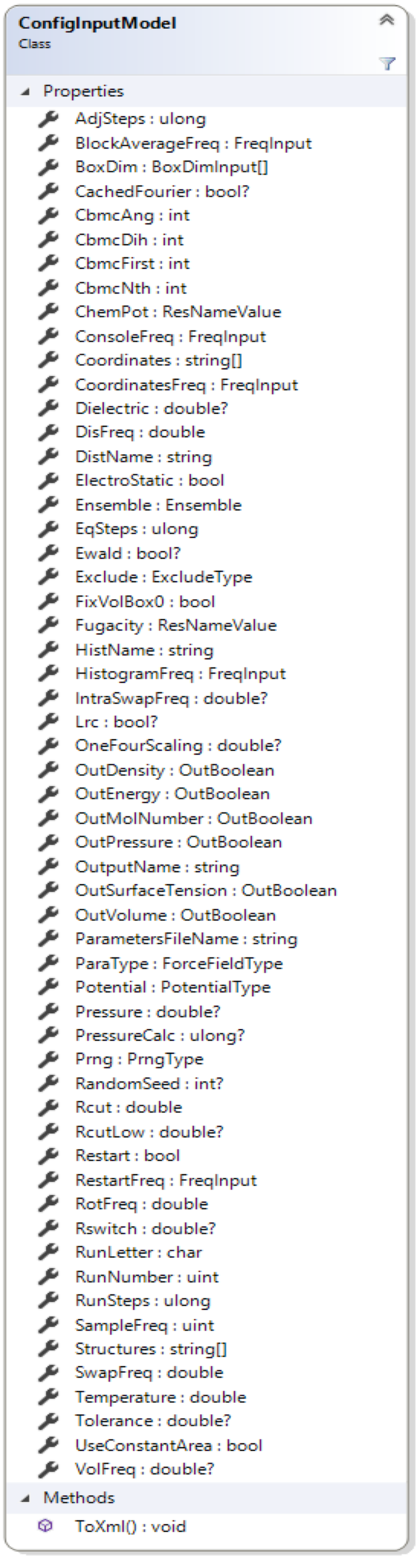


4.6 Class Diagram

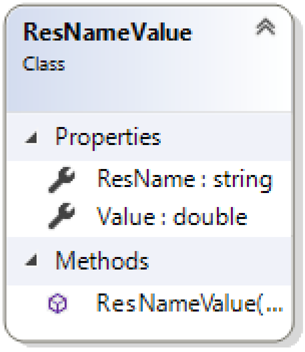
Due to the size of the class diagram, it could not be placed in this document. To view it online, use the link below:
https://drive.google.com/open?id=0B8KjTyY9P8pKMENBcFVtelVITzg

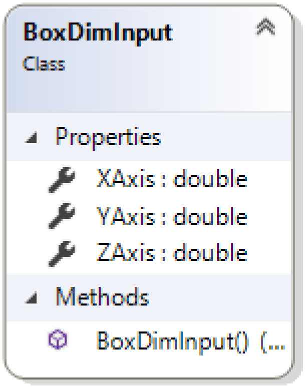
Class List

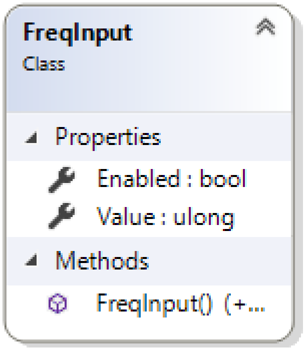
In the next few pages, a list of classes will be shown.

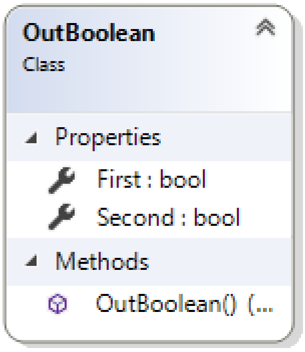


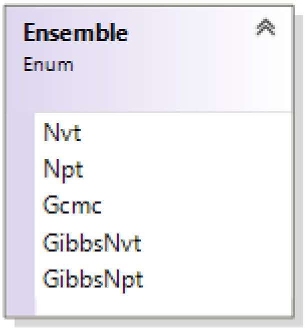
```
ConfigInputModel
Class

Properties
AdjSteps : ulong
BlockAverageFreq : FreqInput
BoxDim : BoxDimInput[]
CachedFourier : bool?
CbmcAng : int
CbmcDih : int
CbmcFirst : int
CbmcNth : int
ChemPot : ResNameValue
ConsoleFreq : FreqInput
Coordinates : string[]
CoordinatesFreq : FreqInput
Dielectric : double?
DisFreq : double
DistName : string
ElectroStatic : bool
Ensemble : Ensemble
EqSteps : ulong
Ewald : bool?
Exclude : ExcludeType
FixVolBox0 : bool
Fugacity : ResNameValue
HistName : string
HistogramFreq : FreqInput
IntraSwapFreq : double?
Lrc : bool?
OneFourScaling : double?
OutDensity : OutBoolean
OutEnergy : OutBoolean
OutMolNumber : OutBoolean
OutPressure : OutBoolean
OutputName : string
OutSurfaceTension : OutBoolean
OutVolume : OutBoolean
ParametersFileName : string
ParaType : ForceFieldType
Potential : PotentialType
Pressure : double?
PressureCalc : ulong?
Prng : PrngType
RandomSeed : int?
Rcut : double
RcutLow : double?
Restart : bool
RestartFreq : FreqInput
RotFreq : double
Rswitch : double?
RunLetter : char
RunNumber : uint
RunSteps : ulong
SampleFreq : uint
Structures : string[]
SwapFreq : double
Temperature : double
Tolerance : double?
UseConstantArea : bool
VolFreq : double?

Methods
ToXml() : void
```

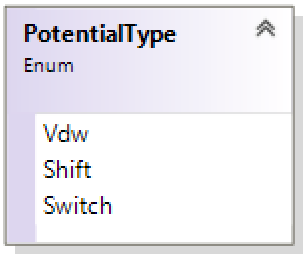

ResNameValue
Class
Properties
ResName : string
Value : double
Methods
ResNameValue(...

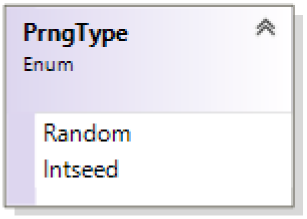

BoxDimInput
Class
Properties
XAxis : double
YAxis : double
ZAxis : double
Methods
BoxDimInput() (...

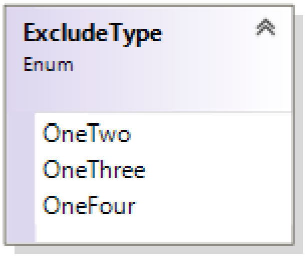

FreqInput
Class
Properties
Enabled : bool
Value : ulong
Methods
FreqInput() (+...

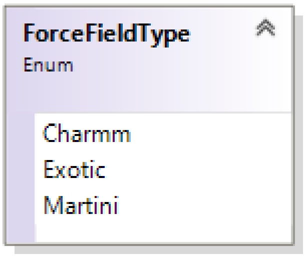

OutBoolean
Class
Properties
First : bool
Second : bool
Methods
OutBoolean() (...

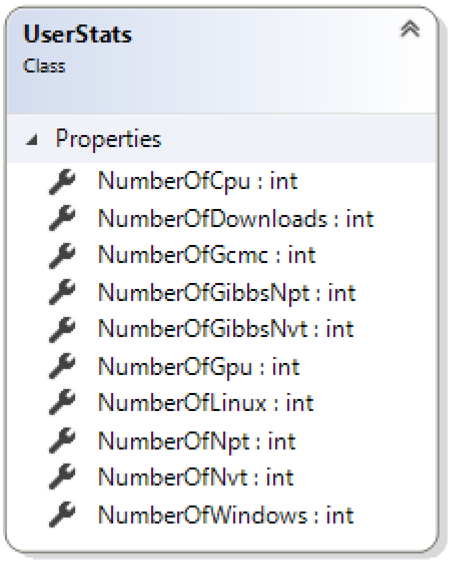

Ensemble
Enum
Nvt
Npt
Gcmc
GibbsNvt
GibbsNpt


PotentialType
Enum
Vdw
Shift
Switch


PrngType
Enum
Random
Intseed


ExcludeType
Enum
OneTwo
OneThree
OneFour


ForceFieldType
Enum
Charmm
Exotic
Martini



NumberOfCpu : int
NumberOfDownloads : int
NumberOfGcmc : int
NumberOfGibbsNpt : int
NumberOfGibbsNvt : int
NumberOfGpu : int
NumberOfLinux : int
NumberOfNpt : int
NumberOfNvt : int
NumberOfWindows : int

4.7 Application Program Interfaces

Registration Api

Endpoint: /api/Registration/Input

Method: GET

Input Example:

```
{
        “userName” : “Ahmed Taher”,
        “userEmail” : “fd3025@wayne.edu”,
        “userAffiliation” : “Wayne State”,
        “extraComment” : “none”,
        “G-recaptcha-response” : “9a3lkjng9032kalkjgaid23”
}
```

Response Example:

```
{
        “Success” : true,
        “Model” : {
                “Name” : “Ahmed Taher”,
                “Email” : “fd3025@wayne.edu”,
                    “Affiliation” : “Wayne State”,
                    “Text” : “none”
            },
    “Errors” : []
}
```

Config Input Api

Endpoint: /api/ConfigInput/FormPost
Method: POST
Input Example:

```
{
    “Ensemble” : “Gcmc”,
    “Restart” : false,
    “PrngType” : “Random”
    ....
}
```

Note: *For a complete list of fields for the input example, refer to the ConfigInputModel class diagram*.

Input Invalid Example:

```
{
    ...
    “Restart” : 5000,
    ...
}
```

Response Valid Example:
File download of “input.xml” which is the XML form of ConfigInputModel.

Response Invalid Example:

On input error, the response will have a status code of 500 as well as JSON content that lists all error fields and cause. An example is provided below:

```
{
        “Restart” : “Could not convert input in to type ‘bool’.”
}
```

Login Api

Endpoint: /api/Login/ValidateLogin

Method: POST

Example Valid Input:

```
{
        “uName” : “gomc@gomc.com”,
        “pCode” : “password123”
}
```

Example Invalid Input:

```
{
        “uName” : “fake@email.com”,
        “pCode” : “z”
}
```

Example Valid Output:

```
{
        “Session” : “7aec529b-5c32-4ead-9715-65e865b83702”
}
```

Example Invalid Output:

On input error, the response will have a status code of 401 as well as message content that describe the reason the input is invalid:

```
{
        “Message” : “The email/password was not found in the database.”
}
```

Endpoint: /api/Login/ValidateSession

Method: POST

Example Input:

```
{
        “session” : “7aec529b-5c32-4ead-9715-65e865b83702”
}
```

Example Output:

```
{
        “Result” : true
}
```

Latex Api

Endpoint: /api/Latex/Convert

Method: POST

Example Input:

```
{
        “file” : “(see note)”,
        “version” : “v2.1”,
        “session” : “7aec529b-5c32-4ead-9715-65e865b83702”
}
```

Note: The value of ‘file’ is the text contents of a LaTeX file.

Example Output:

```
{
        “Result” : “Success”
}
```

Result can be one of the following:

- Success – The request has succeeded in uploading the latex file to the database.
- SessionExpired – The session supplied is valid but has expired.
- InvalidFormat – The supplied file is not a valid latex file.
- BadSession – The session supplied is invalid.

New Announcement Api

Endpoint: /api/Admin/NewAnnounment

Method: POST

Example Valid Input:

```
{
        “Content” : “I am a new announcement message”,
        “Session” : “7aec529b-5c32-4ead-9715-65e865b83702”
}
```

Example Invalid Input:

```
{
        “Content” : 1024,
        “Session” : “I am a session”
}
```

Example Output:

```
{
        “result” : “Success”
}
```

Result can be one of the following:

- Success – The request has succeeded in saving the new announcement to the database.
- SessionExpired – The session supplied is valid but has expired.
- InvalidInput – The input supplied is either missing or is not of correct type (string).
- BadSession – The session supplied is invalid.

GitHub Web-hooks Event Api

Endpoint: /api/Github/Callback

Method: POST

The website will utilize web-hooks to handle specific events. The endpoint for event callback will be ‘/api/Github/Callback’. The callback will respond with an OK (200) status and the content will be the same as the input. The request and response to each web-hooks event sent can be tracked at Github.com (including the content).

The push event is used for the latex repo. When an admin pushes a commit to the latex repo, Github will send that event to our endpoint and the back-end will update the LatexUploads database table. More information about the push event can be found at: https://developer.github.com/v3/activity/events/types/#pushevent.

The releases event will also be used for a stretch goal. Currently, the downloads page gets a JSON of the download links from the GitHub Releases Api when the user requests the downloads page. The stretch goal is to have a web-hook for the releases event. This event occurs whenever there is a new release in the GOMC repo. The back-end will respond by saving the same JSON input as before into the database. The downloads page will pull that JSON from the database, instead of getting it on-demand from Github every page request. More information can be obtained about the releases event at: https://developer.github.com/v3/activity/events/types/#releaseevent.

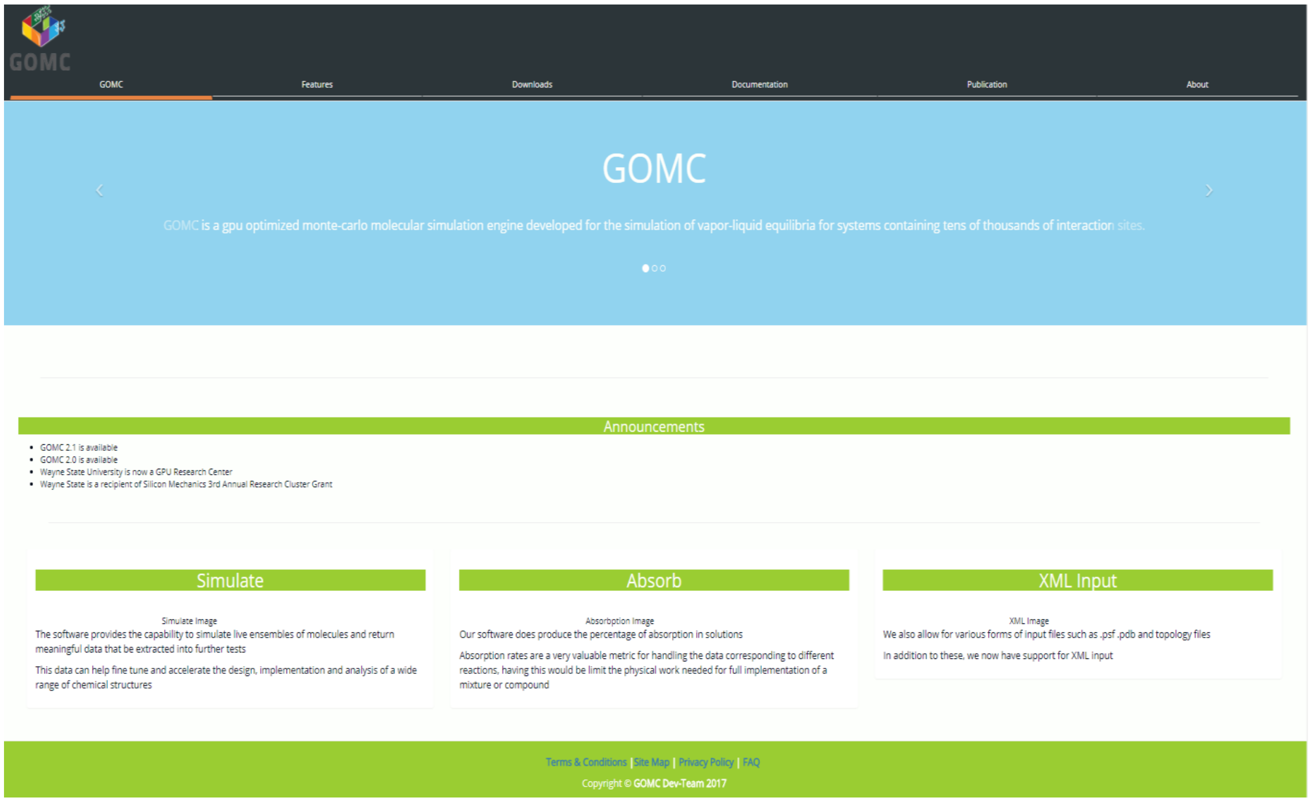
4.8 User Interface Design

The images below correspond to the current UI design that is in development and will be updated as we continue to make changes and add in new components. The previous version consisted of nine static pages that can be viewed at http://gomc.eng.wayne.edu.

The new site can be found currently at http://ahtaher.net/. As of the time of writing this document, the site has eleven pages with two dynamic components namely being the downloads list and the examples list.

Each of the figures below will show a page and talk briefly about its functionality.

Note: All areas in the below section marked with an asterisk (*) are areas that describe functionality that is different from what the current screenshot shows. These features are currently under-development or have currently opened issues associated with them which are being tracked via GitHub issues here: https://github.com/WSU-Capstone-2017/GOMC-Website/issues



GOMC

GOMC is a gpu optimized monte-carlo molecular simulation engine developed for the simulation of vapor-liquid equilibria for systems containing tens of thousands of interaction sites.

Announcements

- GOMC 2.1 is available
- GOMC 2.0 is available
- Wayne State University is now a GPU Research Center
- Wayne State is a recipient of Silicon Mechanics 3rd Annual Research Cluster Grant

Simulate

Simulate Image

The software provides the capability to simulate live ensembles of molecules and return meaningful data that be extracted into further tests

This data can help fine tune and accelerate the design, implementation and analysis of a wide range of chemical structures

Absorb

Absorbption Image

Our software does produce the percentage of absorption in solutions

Absorption rates are a very valuable metric for handling the data corresponding to different reactions, having this would be limit the physical work needed for full implementation of a mixture or compound

XML Input

XML Image

We also allow for various forms of input files such as .psf .pdb and topology files

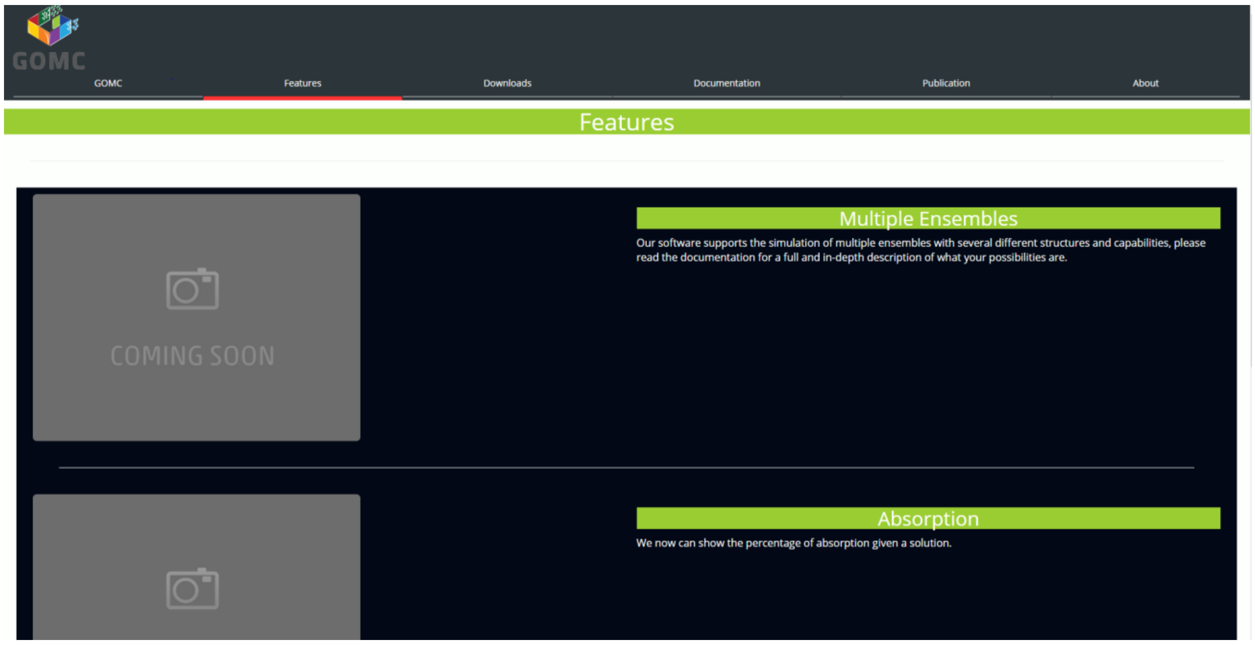
In addition to these, we now have support for XML input

Terms & Conditions | Site Map | Privacy Policy | FAQ

Copyright © GOMC Dev-Team 2017

*** Figure A: The landing page***

- Main landing page for user will contain a jumbotron carousel with the following data:
    - A short blurb about what GOMC is
    - A link to the downloads page featuring the most recent release
    - A link to the documentation page
- A 1x3 row that discusses the main features to market the newest release
- * Announcements content should be editable from the Admin view



Features

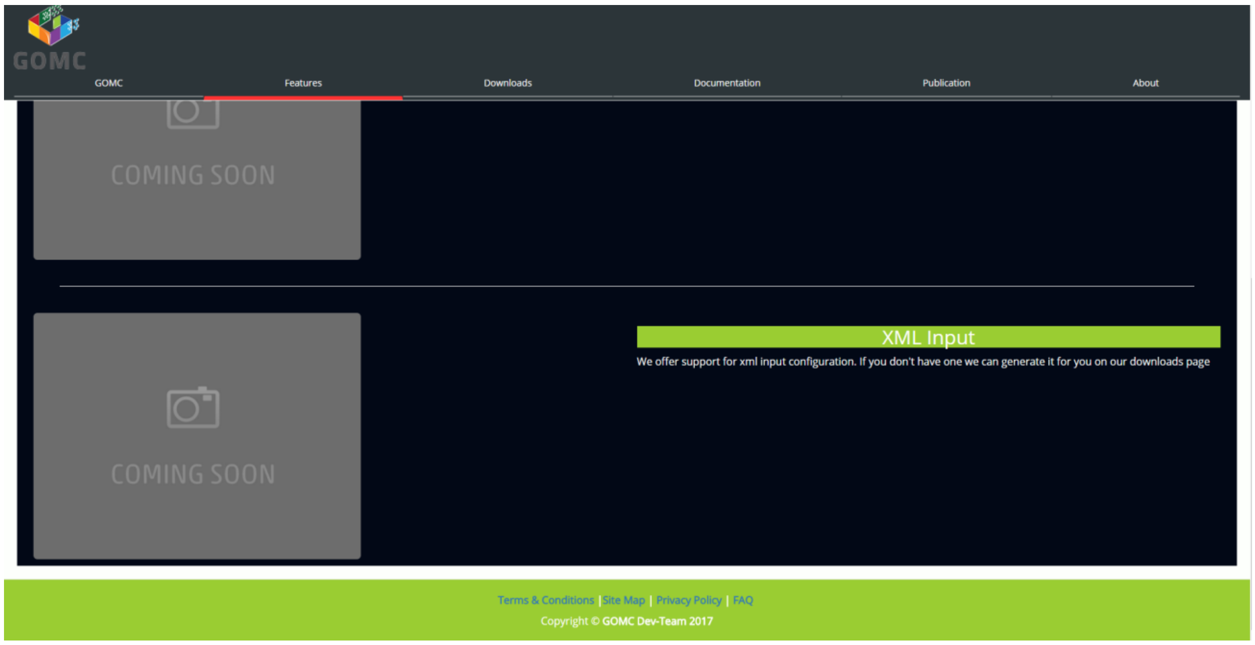
Multiple Ensembles

Our software supports the simulation of multiple ensembles with several different structures and capabilities, please read the documentation for a full and in-depth description of what your possibilities are.

Absorption

We now can show the percentage of absorption given a solution.

Figure B: Features Page, Top-Half



XML Input

We offer support for xml input configuration. If you don't have one we can generate it for you on our downloads page

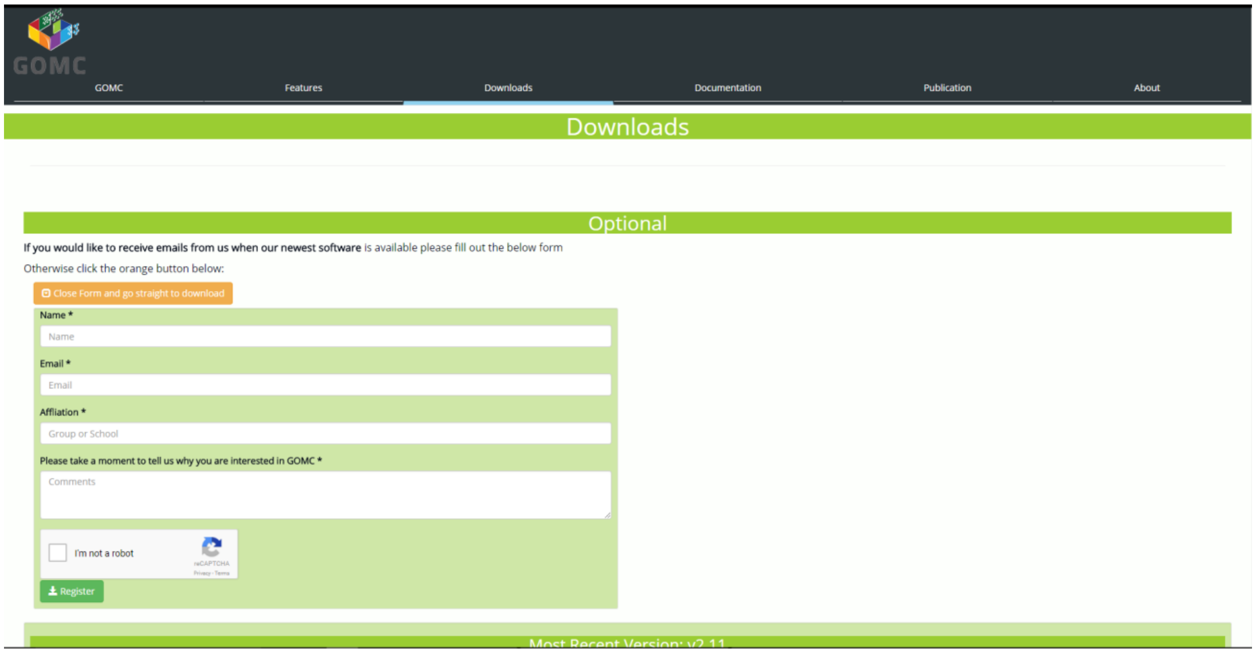
Terms & Conditions | Site Map | Privacy Policy | FAQ

Copyright © GOMC Dev-Team 2017

** Figure C: Features Page, Bottom-Half*

- Second tab will have the goal to be a highlight of several features that the GOMC software would like to display
- * Content will be arranged into 1x3 rows of cards about each feature, each one displaying the following:

- An image
- A header
- A one to two-line paragraph elaborating on each feature



Downloads

Optional

If you would like to receive emails from us when our newest software is available please fill out the below form

Otherwise click the orange button below:

Close Form and go straight to download

Name *

Name

Email *

Email

Affliation *

Group or School

Please take a moment to tell us why you are interested in GOMC *

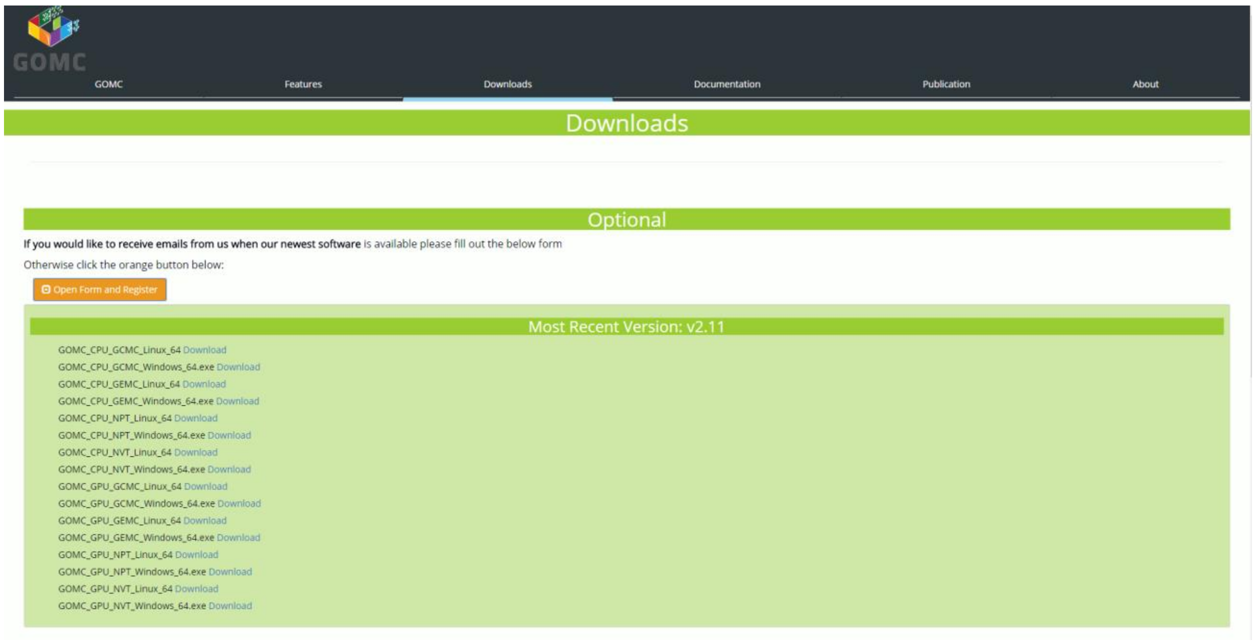
Comments

I'm not a robot

Register

Most Recent Version: v2.11

Figure D: Downloads Page, Top-Half, Registration Form Open



Downloads

Optional

If you would like to receive emails from us when our newest software is available please fill out the below form

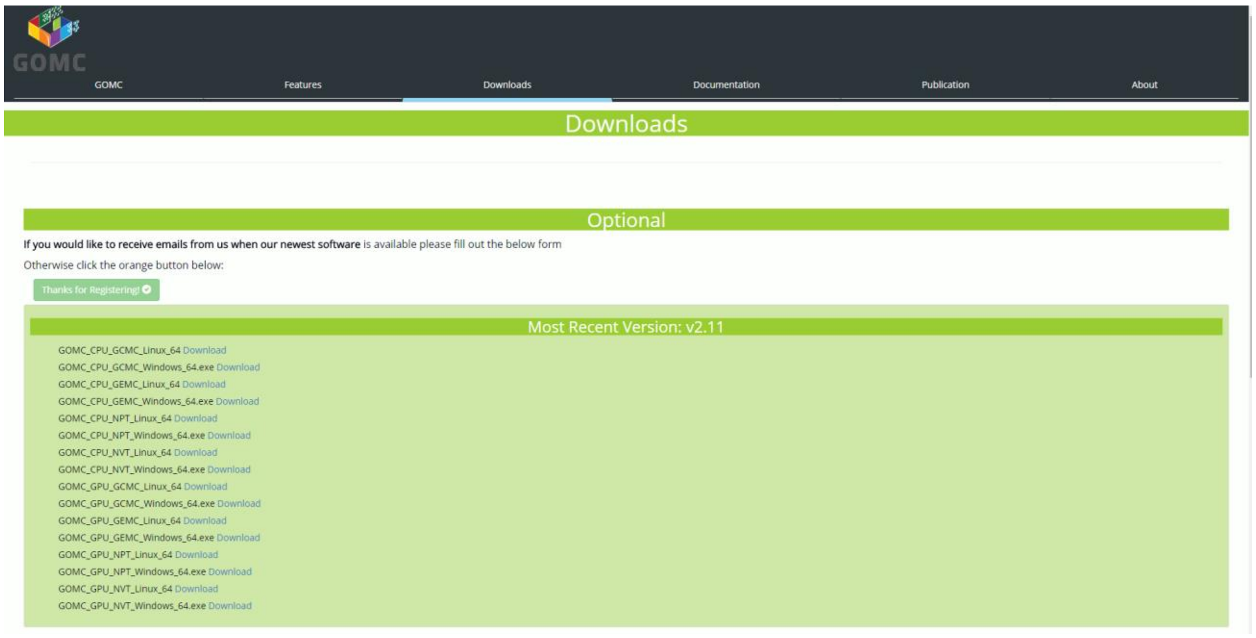
Otherwise click the orange button below:

Open Form and Register

Most Recent Version: v2.11

GOMC_CPU_GCMC_Linux_64 Download
GOMC_CPU_GCMC_Windows_64.exe Download
GOMC_CPU_GEMC_Linux_64 Download
GOMC_CPU_GEMC_Windows_64.exe Download
GOMC_CPU_NPT_Linux_64 Download
GOMC_CPU_NPT_Windows_64.exe Download
GOMC_CPU_NVT_Linux_64 Download
GOMC_CPU_NVT_Windows_64.exe Download
GOMC_GPU_GCMC_Linux_64 Download
GOMC_GPU_GCMC_Windows_64.exe Download
GOMC_GPU_GEMC_Linux_64 Download
GOMC_GPU_GEMC_Windows_64.exe Download
GOMC_GPU_NPT_Linux_64 Download
GOMC_GPU_NPT_Windows_64.exe Download
GOMC_GPU_NVT_Linux_64 Download
GOMC_GPU_NVT_Windows_64.exe Download

Figure E: Downloads Page, Top-Half, Registration Form Closed



Downloads

Optional

If you would like to receive emails from us when our newest software is available please fill out the below form

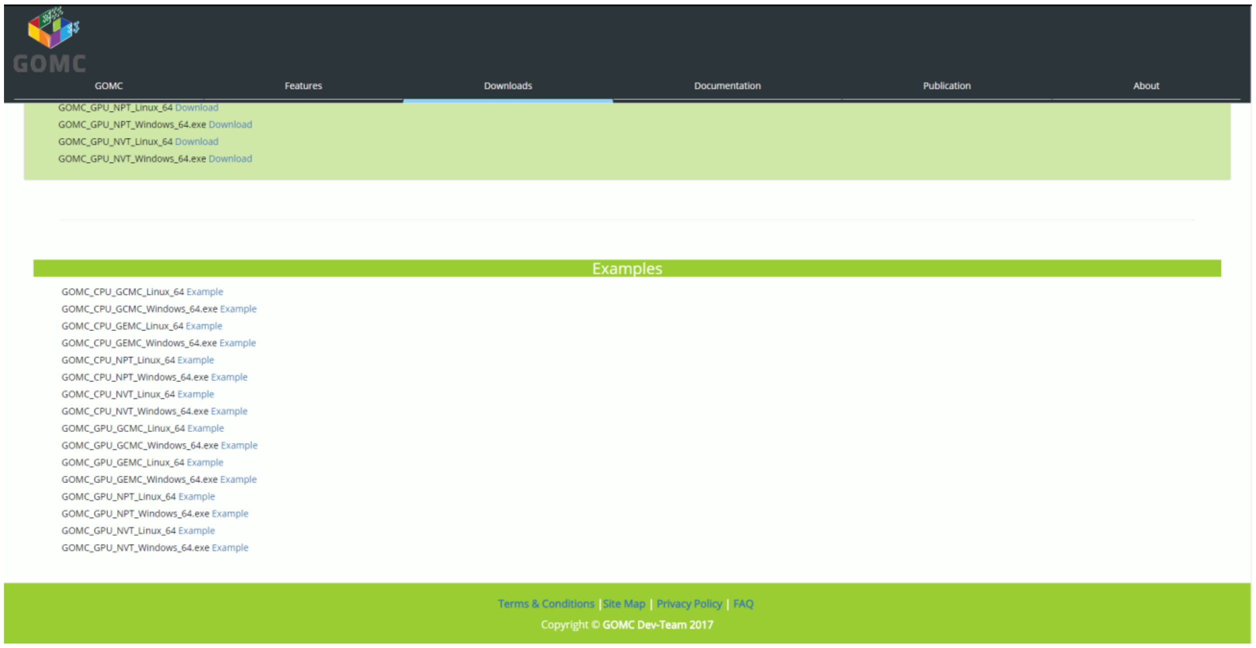
Otherwise click the orange button below:

Most Recent Version: v2.11

GOMC_CPU_GCMC_Linux_64 Download
GOMC_CPU_GCMC_Windows_64.exe Download
GOMC_CPU_GEMC_Linux_64 Download
GOMC_CPU_GEMC_Windows_64.exe Download
GOMC_CPU_NPT_Linux_64 Download
GOMC_CPU_NPT_Windows_64.exe Download
GOMC_CPU_NVT_Linux_64 Download
GOMC_CPU_NVT_Windows_64.exe Download
GOMC_GPU_GCMC_Linux_64 Download
GOMC_GPU_GCMC_Windows_64.exe Download
GOMC_GPU_GEMC_Linux_64 Download
GOMC_GPU_GEMC_Windows_64.exe Download
GOMC_GPU_NPT_Linux_64 Download
GOMC_GPU_NPT_Windows_64.exe Download
GOMC_GPU_NVT_Linux_64 Download
GOMC_GPU_NVT_Windows_64.exe Download

Figure F: Downloads Page, Top-Half, Registration Form Submitted

- Downloads page contains a form that the user can optionally fill out with data that way the user will be added to the GOMC mailing list
- There must be a button that optionally opens and closes the form
- After the user fills out the form, it should be closed, and must give the user a feedback message letting them know that their submission was complete
- The form should then be locked and the button to open the form again should be disabled



GOMC_GPU_NPT_Linux_64 Download
GOMC_GPU_NPT_Windows_64.exe Download
GOMC_GPU_NVT_Linux_64 Download
GOMC_GPU_NVT_Windows_64.exe Download

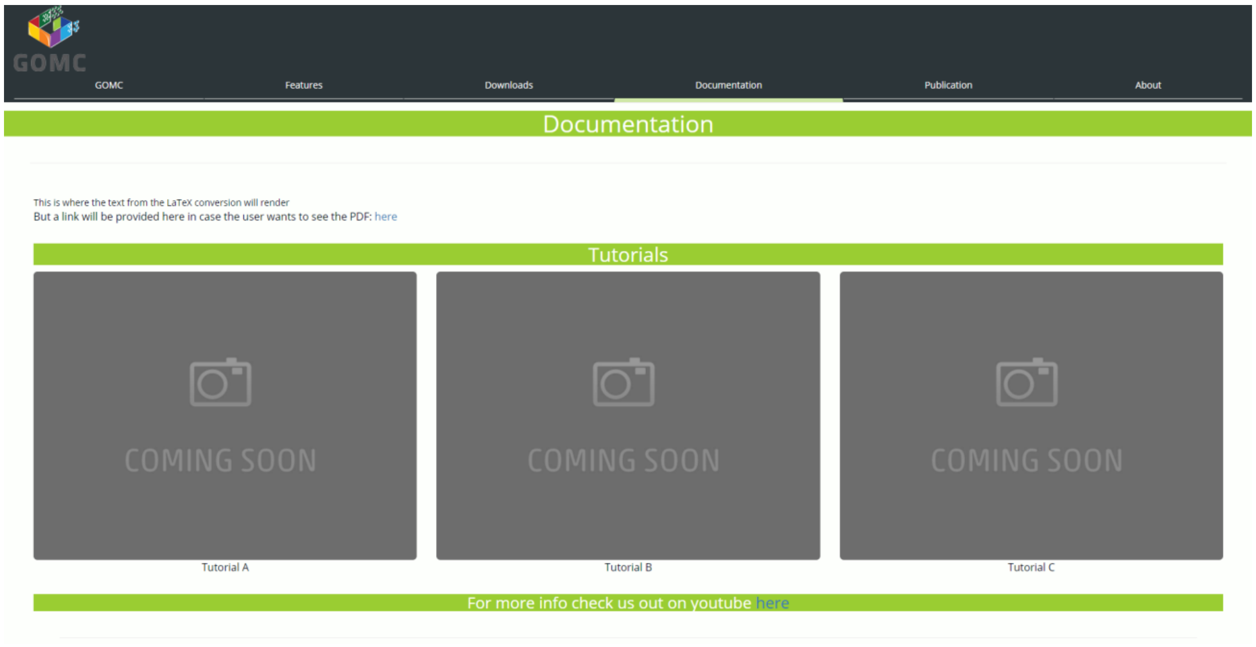
Examples

GOMC_CPU_GCMC_Linux_64 Example
GOMC_CPU_GCMC_Windows_64.exe Example
GOMC_CPU_GEMC_Linux_64 Example
GOMC_CPU_GEMC_Windows_64.exe Example
GOMC_CPU_NPT_Linux_64 Example
GOMC_CPU_NPT_Windows_64.exe Example
GOMC_CPU_NVT_Linux_64 Example
GOMC_CPU_NVT_Windows_64.exe Example
GOMC_GPU_GCMC_Linux_64 Example
GOMC_GPU_GCMC_Windows_64.exe Example
GOMC_GPU_GEMC_Linux_64 Example
GOMC_GPU_GEMC_Windows_64.exe Example
GOMC_GPU_NPT_Linux_64 Example
GOMC_GPU_NPT_Windows_64.exe Example
GOMC_GPU_NVT_Linux_64 Example
GOMC_GPU_NVT_Windows_64.exe Example

Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure G: Downloads Page, Lower-Half

- The downloads list and examples list must be dynamically generated from the back-end C# layer with the data returned from a *Get* response from the GitHub API
    - The data must be formatted into title and link
    - The link must return the appropriate data being executables for the downloads and zips for the examples



This is where the text from the LaTeX conversion will render
But a link will be provided here in case the user wants to see the PDF: here

For more info check us out on youtube here

Figure H: Documentation Page, Top-Half

- The documentation page will have the following:
  - * Two buttons for pdf view and html view
  - Tutorial cards will be built in a similar manner to features but with links to YouTube videos showing setup, running examples, and other parts
  - A link to the YouTube channel

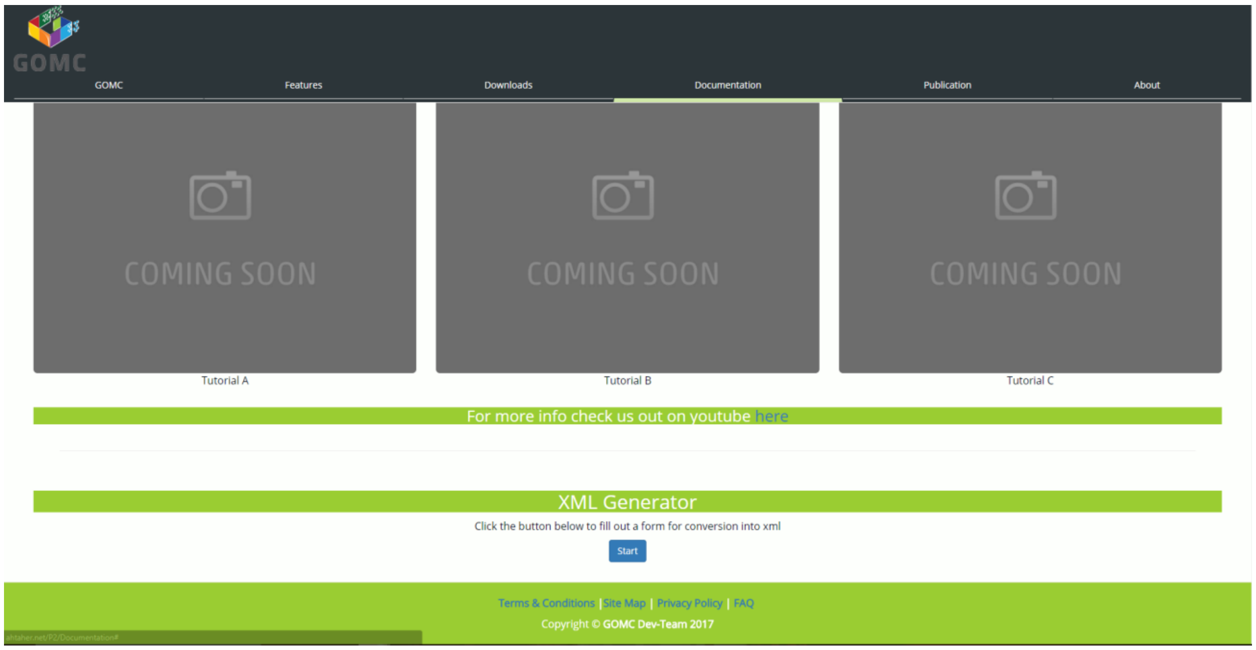

GOMC
GOMC
Features
Downloads
Documentation
Publication
About
COMING SOON
COMING SOON
COMING SOON
Tutorial A
Tutorial B
Tutorial C
For more info check us out on youtube here
XML Generator
Click the button below to fill out a form for conversion into xml
Start
Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure I: Documentation Page, Lower-Half, XML-Generator closed

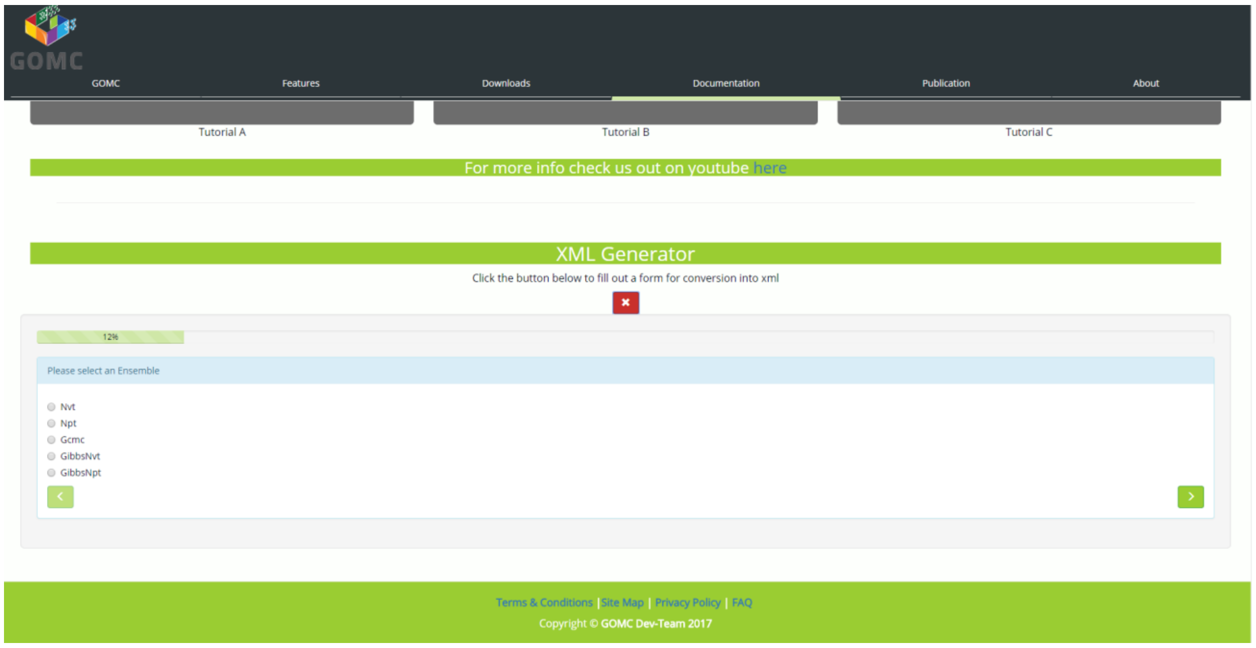
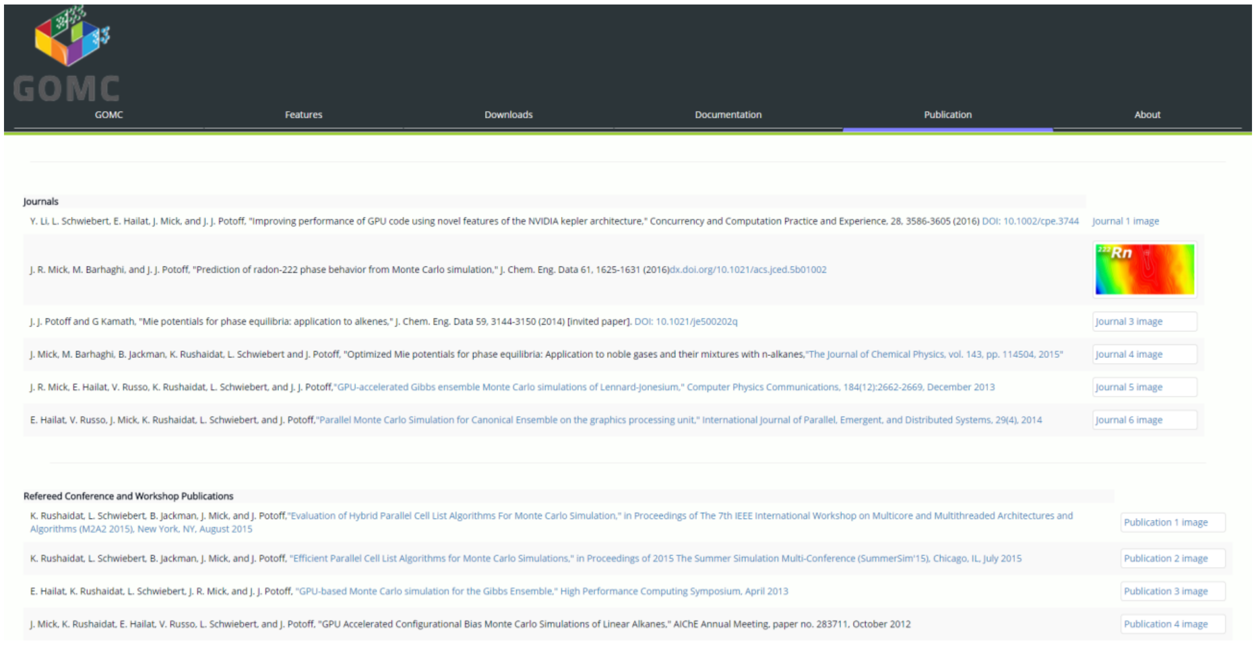

GOMC
GOMC
Features
Downloads
Documentation
Publication
About
Tutorial A
Tutorial B
Tutorial C
For more info check us out on youtube here
XML Generator
Click the button below to fill out a form for conversion into xml
12%
Please select an Ensemble
Nvt
Npt
Gcmc
GibbsNvt
GibbsNpt
Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure J: XML-Generator form opened

- The XML generator will present a form that the user can fill out with appropriate data
- This data will then be sent to an endpoint by a *POST* request which will then be processed and will return the XML download to the user

- * The XML generator button should lead to its own subpage



Journals

Y. Li, L. Schwiebert, E. Hailat, J. Mick, and J. J. Potoff, "Improving performance of GPU code using novel features of the NVIDIA kepler architecture," Concurrency and Computation Practice and Experience, 28, 3586-3605 (2016) DOI: 10.1002/cpe.3744 | Journal 1 image

J. R. Mick, M. Barhaghi, and J. J. Potoff, "Prediction of radon-222 phase behavior from Monte Carlo simulation," J. Chem. Eng. Data 61, 1625-1631 (2016)dx.doi.org/10.1021/acs.jced.5b01002

J. J. Potoff and G Kamath, "Mie potentials for phase equilibria: application to alkenes," J. Chem. Eng. Data 59, 3144-3150 (2014) [invited paper]. DOI: 10.1021/je500202q | Journal 3 image

J. Mick, M. Barhaghi, B. Jackman, K. Rushaidat, L. Schwiebert and J. Potoff, "Optimized Mie potentials for phase equilibria: Application to noble gases and their mixtures with n-alkanes,"The Journal of Chemical Physics, vol. 143, pp. 114504, 2015" | Journal 4 image

J. R. Mick, E. Hailat, V. Russo, K. Rushaidat, L. Schwiebert, and J. J. Potoff,"GPU-accelerated Gibbs ensemble Monte Carlo simulations of Lennard-Jonesium," Computer Physics Communications, 184(12):2662-2669, December 2013 | Journal 5 image

E. Hailat, V. Russo, J. Mick, K. Rushaidat, L. Schwiebert, and J. Potoff,"Parallel Monte Carlo Simulation for Canonical Ensemble on the graphics processing unit," International Journal of Parallel, Emergent, and Distributed Systems, 29(4), 2014 | Journal 6 image

Refereed Conference and Workshop Publications

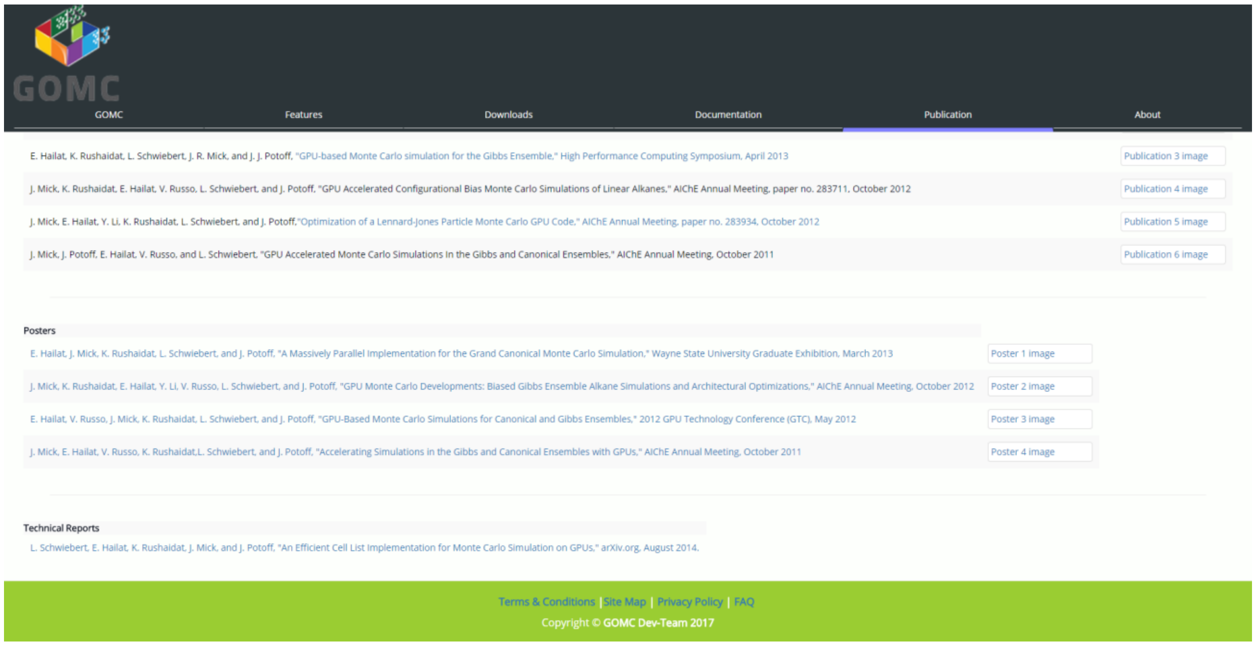
K. Rushaidat, L. Schwiebert, B. Jackman, J. Mick, and J. Potoff,"Evaluation of Hybrid Parallel Cell List Algorithms For Monte Carlo Simulation," in Proceedings of The 7th IEEE International Workshop on Multicore and Multithreaded Architectures and Algorithms (M2A2 2015), New York, NY, August 2015 | Publication 1 image

K. Rushaidat, L. Schwiebert, B. Jackman, J. Mick, and J. Potoff, "Efficient Parallel Cell List Algorithms for Monte Carlo Simulations," in Proceedings of 2015 The Summer Simulation Multi-Conference (SummerSim'15), Chicago, IL, July 2015 | Publication 2 image

E. Hailat, K. Rushaidat, L. Schwiebert, J. R. Mick, and J. J. Potoff, "GPU-based Monte Carlo simulation for the Gibbs Ensemble," High Performance Computing Symposium, April 2013 | Publication 3 image

J. Mick, K. Rushaidat, E. Hailat, V. Russo, L. Schwiebert, and J. Potoff, "GPU Accelerated Configurational Bias Monte Carlo Simulations of Linear Alkanes," AIChE Annual Meeting, paper no. 283711, October 2012 | Publication 4 image

Figure K: Publications, Top-Half



E. Hailat, K. Rushaidat, L. Schwiebert, J. R. Mick, and J. J. Potoff, "GPU-based Monte Carlo simulation for the Gibbs Ensemble," High Performance Computing Symposium, April 2013 | Publication 3 image

J. Mick, K. Rushaidat, E. Hailat, V. Russo, L. Schwiebert, and J. Potoff, "GPU Accelerated Configurational Bias Monte Carlo Simulations of Linear Alkanes," AIChE Annual Meeting, paper no. 283711, October 2012 | Publication 4 image

J. Mick, E. Hailat, Y. Li, K. Rushaidat, L. Schwiebert, and J. Potoff,"Optimization of a Lennard-Jones Particle Monte Carlo GPU Code," AIChE Annual Meeting, paper no. 283934, October 2012 | Publication 5 image

J. Mick, J. Potoff, E. Hailat, V. Russo, and L. Schwiebert, "GPU Accelerated Monte Carlo Simulations In the Gibbs and Canonical Ensembles," AIChE Annual Meeting, October 2011 | Publication 6 image

Posters

E. Hailat, J. Mick, K. Rushaidat, L. Schwiebert, and J. Potoff, "A Massively Parallel Implementation for the Grand Canonical Monte Carlo Simulation," Wayne State University Graduate Exhibition, March 2013 | Poster 1 image

J. Mick, K. Rushaidat, E. Hailat, Y. Li, V. Russo, L. Schwiebert, and J. Potoff, "GPU Monte Carlo Developments: Biased Gibbs Ensemble Alkane Simulations and Architectural Optimizations," AIChE Annual Meeting, October 2012 | Poster 2 image

E. Hailat, V. Russo, J. Mick, K. Rushaidat, L. Schwiebert, and J. Potoff, "GPU-Based Monte Carlo Simulations for Canonical and Gibbs Ensembles," 2012 GPU Technology Conference (GTC), May 2012 | Poster 3 image

J. Mick, E. Hailat, V. Russo, K. Rushaidat,L. Schwiebert, and J. Potoff, "Accelerating Simulations in the Gibbs and Canonical Ensembles with GPUs," AIChE Annual Meeting, October 2011 | Poster 4 image

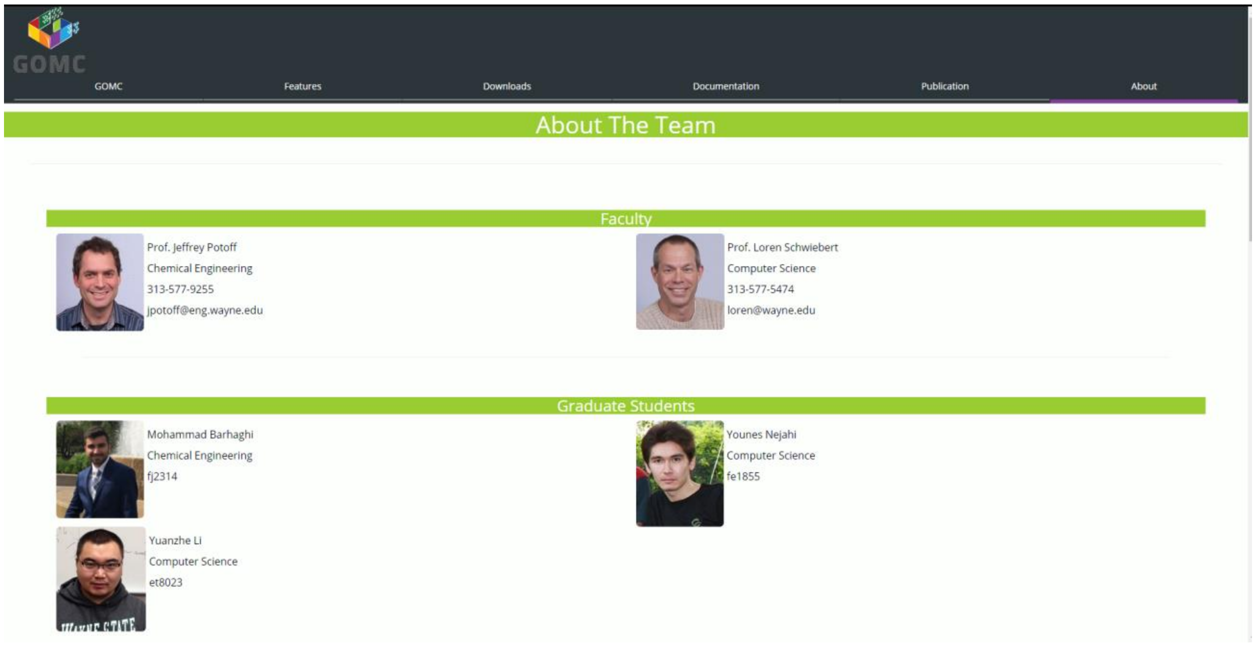
Technical Reports

L. Schwiebert, E. Hailat, K. Rushaidat, J. Mick, and J. Potoff, "An Efficient Cell List Implementation for Monte Carlo Simulation on GPUs," arXiv.org, August 2014.

Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure L: Publications, Lower-Half

- The publications page will have 4 tables for the *Journals*, *Publications*, *Posters and Technical Reports*. This data will present a title and images that correspond to the documents as well as hyperlinks to where they can be found
- * All pages will have images and only the title hyperlinked



About The Team

Faculty

Prof. Jeffrey Potoff
Chemical Engineering
313-577-9255
jpotoff@eng.wayne.edu

Prof. Loren Schwiebert
Computer Science
313-577-5474
loren@wayne.edu

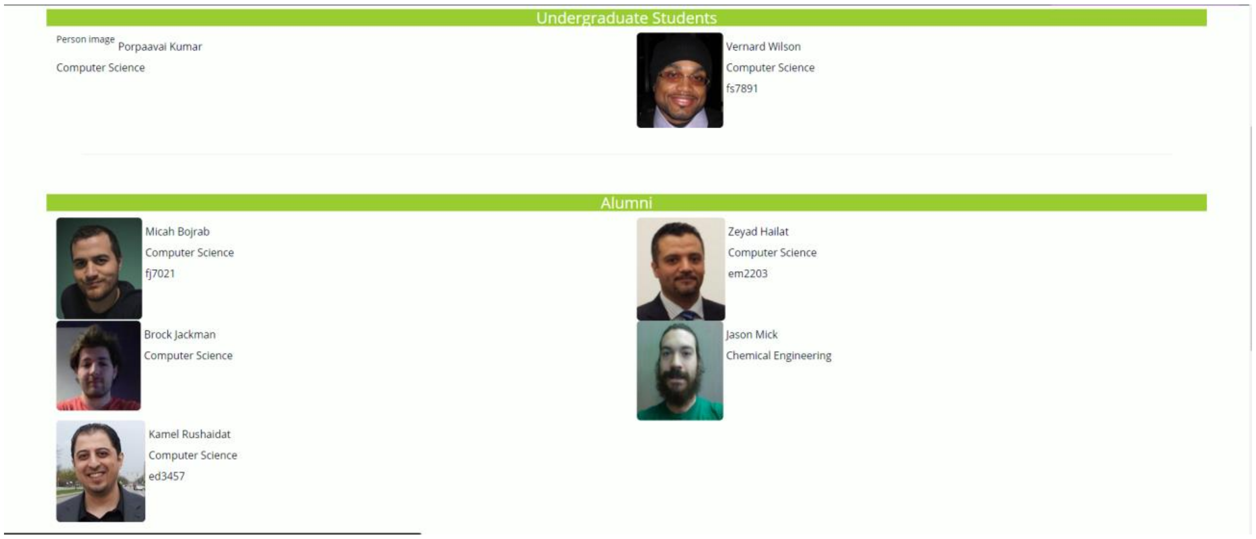
Graduate Students

Mohammad Barhaghi
Chemical Engineering
fj2314

Younes Nejahi
Computer Science
fe1855

Yuanzhe Li
Computer Science
et8023

Figure M: About Page, Top-Part



Undergraduate Students

Person image Porpaavai Kumar
Computer Science

Vernard Wilson
Computer Science
fs7891

Alumni

Micah Bojrab
Computer Science
fj7021

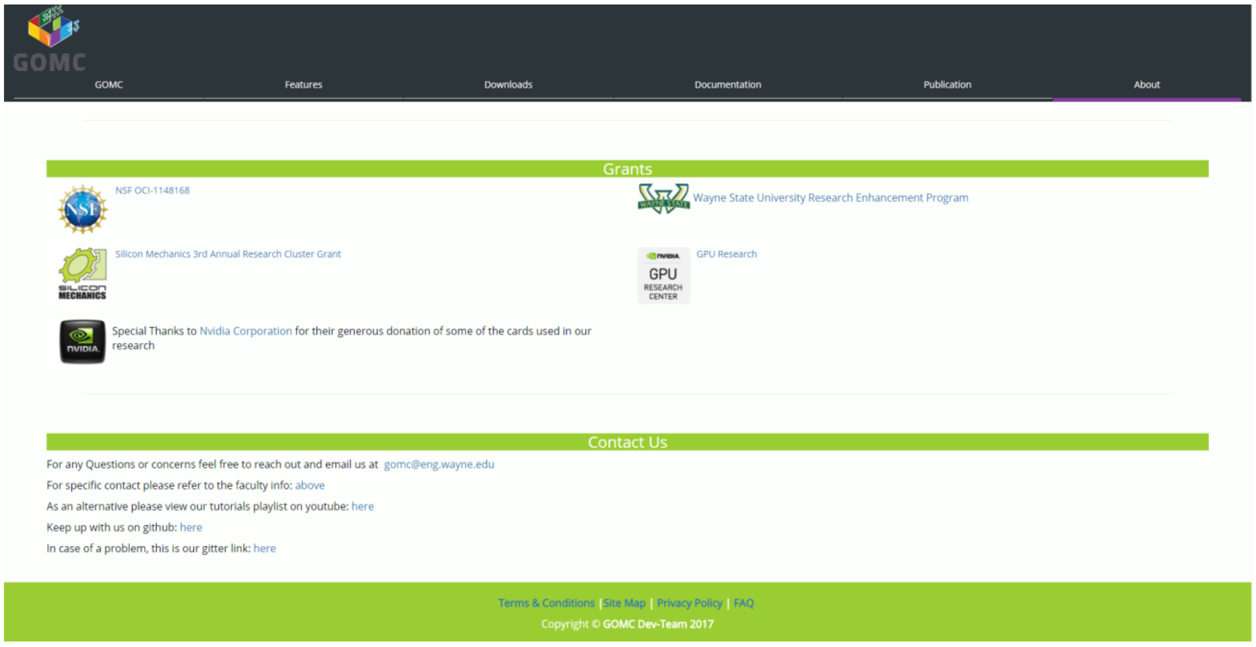
Zeyad Hailat
Computer Science
em2203

Brock Jackman
Computer Science

Jason Mick
Chemical Engineering

Kamel Rushaidat
Computer Science
ed3457

Figure N: About Page, Middle-Part



Grants

NSF OCI-1148168

Wayne State University Research Enhancement Program

Silicon Mechanics 3rd Annual Research Cluster Grant

GPU Research

Special Thanks to Nvidia Corporation for their generous donation of some of the cards used in our research

Contact Us

For any Questions or concerns feel free to reach out and email us at gomc@eng.wayne.edu
For specific contact please refer to the faculty info: above
As an alternative please view our tutorials playlist on youtube: here
Keep up with us on github: here
In case of a problem, this is our gitter link: here

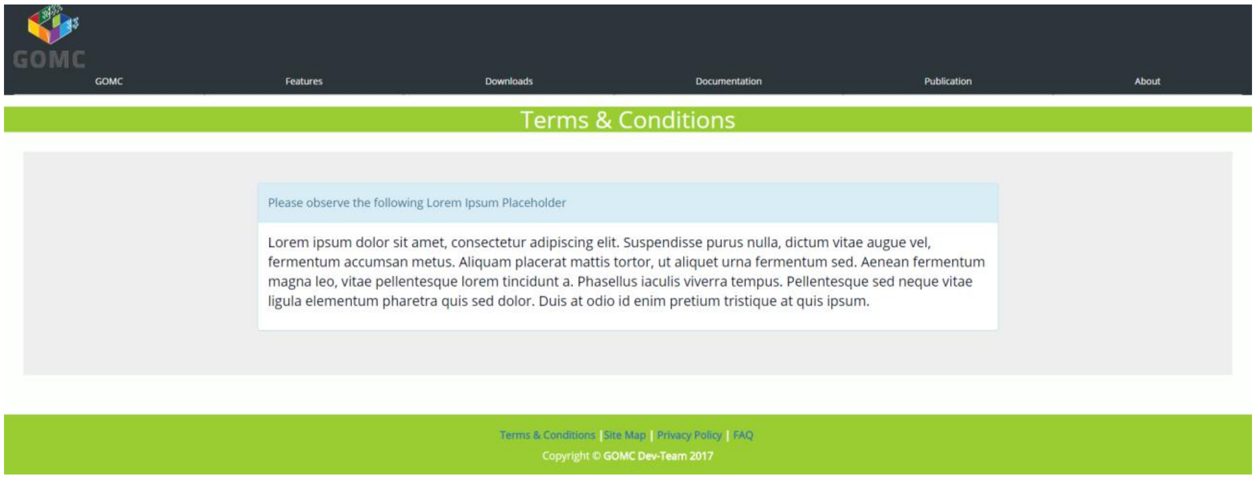
Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure O: About Page, Bottom-Part

- The About page will contain all contributing faculty and student supporters as well as grants
    - All students and alumni will have an image, their name, and their access ID associated

  - All faculty will have their image, name, office phone, and email address added to their card

- The About page will also contain all contact information with respect to the project
  - Email links to the GOMC email
  - Gitter link
  - YouTube link



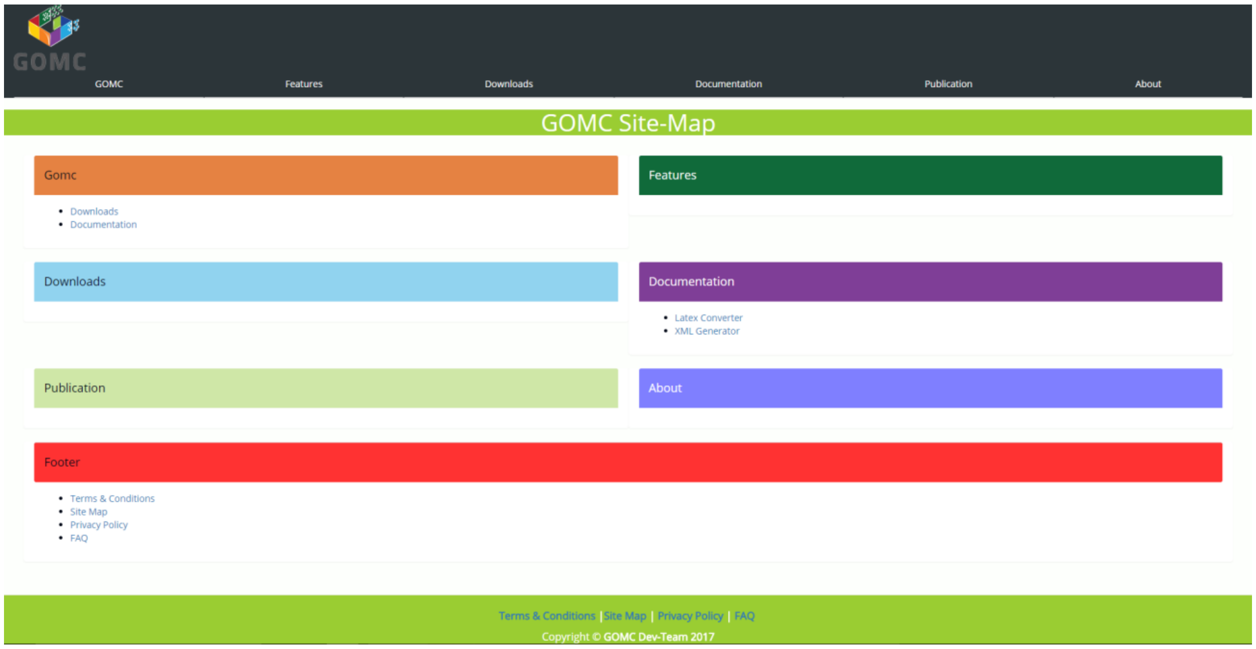
Please observe the following Lorem Ipsum Placeholder

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse purus nulla, dictum vitae augue vel, fermentum accumsan metus. Aliquam placerat mattis tortor, ut aliquet urna fermentum sed. Aenean fermentum magna leo, vitae pellentesque lorem tincidunt a. Phasellus iaculis viverra tempus. Pellentesque sed neque vitae ligula elementum pharetra quis sed dolor. Duis at odio id enim pretium tristique at quis ipsum.

Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure P: Terms and Conditions Page

- There must be a single panel with the header *Terms & Conditions* and the body containing the terms & conditions as provided by the client



GOMC Site-Map

Gomc

- Downloads
- Documentation

Features

Downloads

Documentation

- Latex Converter
- XML Generator

Publication

About

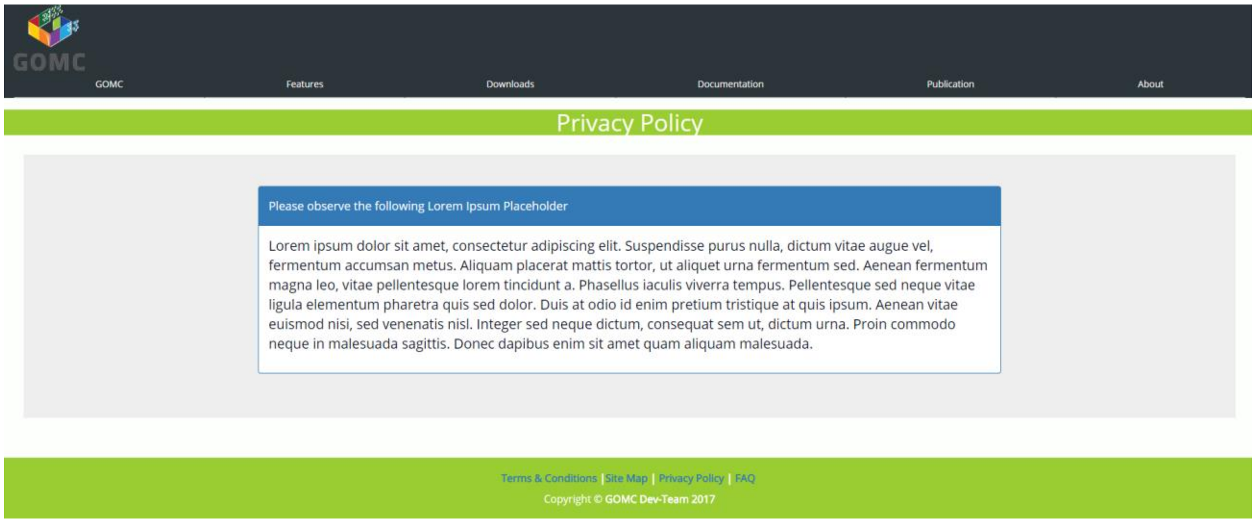
Footer

- Terms & Conditions
- Site Map
- Privacy Policy
- FAQ

Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure Q: Site Map Page

- The site map of the website must have all internal links on each page listed in a tree like format for each page and subpage



Privacy Policy

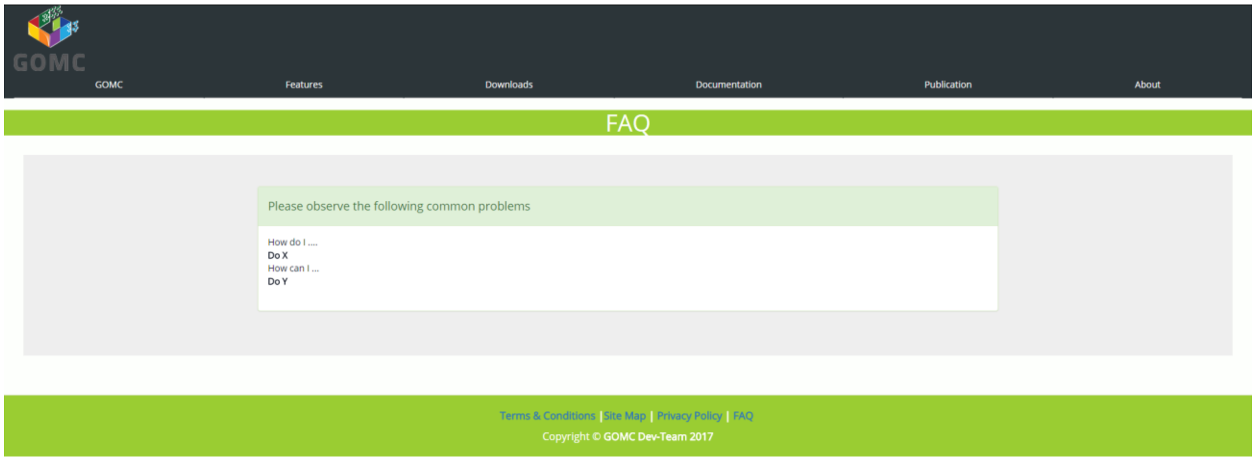
Please observe the following Lorem Ipsum Placeholder

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse purus nulla, dictum vitae augue vel, fermentum accumsan metus. Aliquam placerat mattis tortor, ut aliquet urna fermentum sed. Aenean fermentum magna leo, vitae pellentesque lorem tincidunt a. Phasellus iaculis viverra tempus. Pellentesque sed neque vitae ligula elementum pharetra quis sed dolor. Duis at odio id enim pretium tristique at quis ipsum. Aenean vitae euismod nisi, sed venenatis nisl. Integer sed neque dictum, consequat sem ut, dictum urna. Proin commodo neque in malesuada sagittis. Donec dapibus enim sit amet quam aliquam malesuada.

Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure R: Privacy Policy Page

- There must be a single panel with the header *Privacy Policy* and the body containing the privacy policy as provided by the client



FAQ

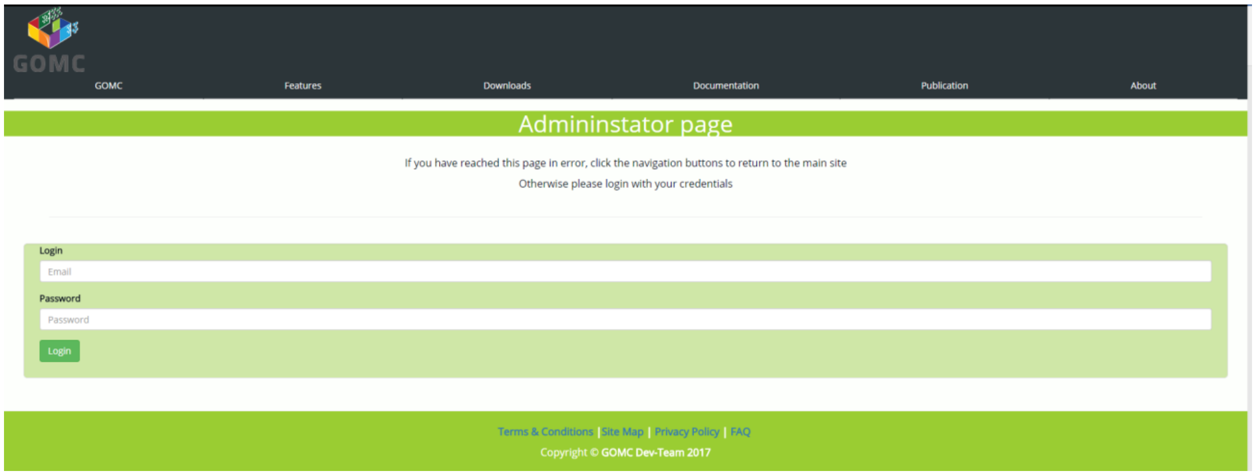
Please observe the following common problems

How do I ....
Do X
How can I ...
Do Y

Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure S: FAQ Page

- The FAQ page must show 1x3 rows of panels
  - Panels must be collapsible
  - Each question should be a panel-header and each answer within its body



Admininstator page

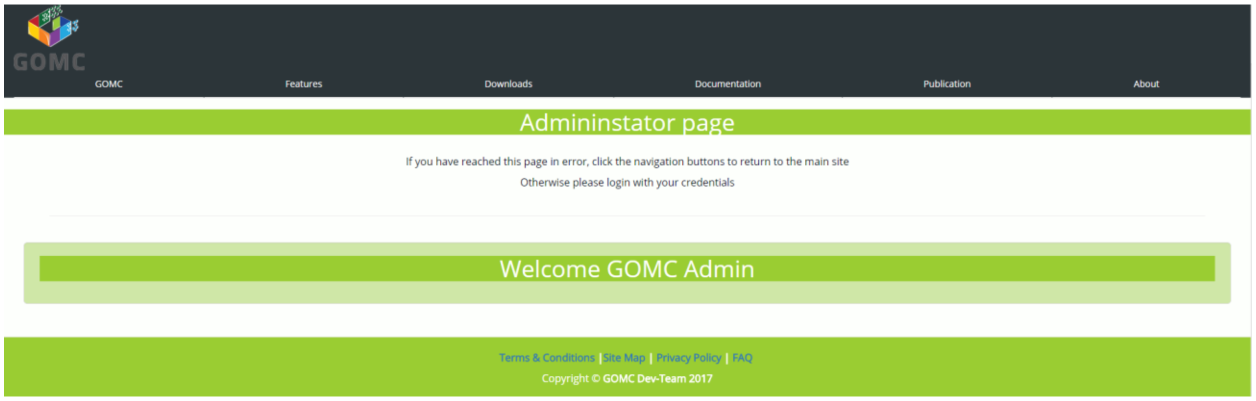
If you have reached this page in error, click the navigation buttons to return to the main site
Otherwise please login with your credentials

Login

Password

Terms & Conditions | Site Map | Privacy Policy | FAQ
Copyright © GOMC Dev-Team 2017

Figure T: Admin Page, Initial



Admininstator page

If you have reached this page in error, click the navigation buttons to return to the main site

Otherwise please login with your credentials

Welcome GOMC Admin

Terms & Conditions | Site Map | Privacy Policy | FAQ

Copyright © GOMC Dev-Team 2017

Figure U Admin Page, Logged in

- The Admin page must have a form with two fields that the user can fill out and submit to login as admin
    - On load, the page must check against the *session storage* to see if a valid GUID exists from the database, if so then the user is already logged in, if not, then the user is not logged in and the view must present the option for them to
    - Upon submission, there should be validation on the front-end to ensure that the login and password meet certain criteria. If they pass, they are sent to the backend for further processing
    - The backend processes the data received and implements our custom *SHA-256* hashing method. If the hash validates, the logged in view as of Figure U is returned

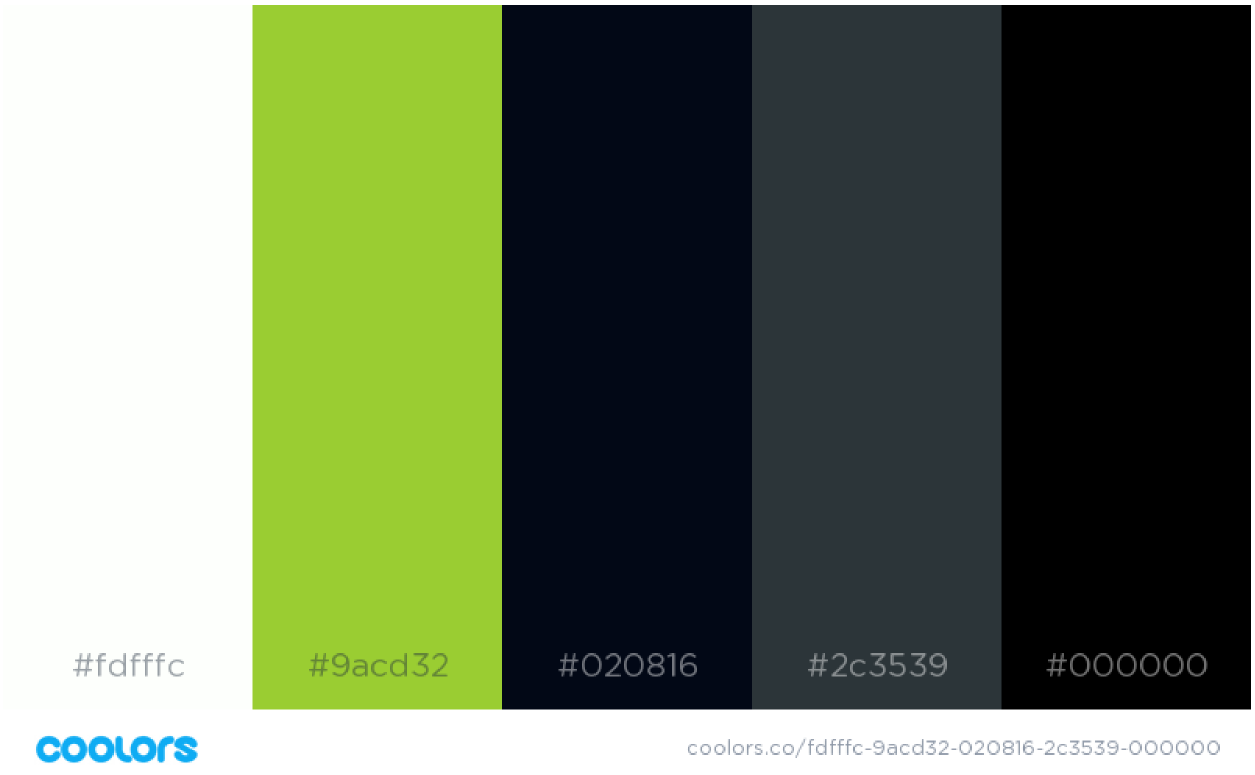


Figure V: Official Pallet

- The main color scheme of the website; these colors will be the main color scheme choices of text color and background color to keep a related look and feel as the user navigates the website

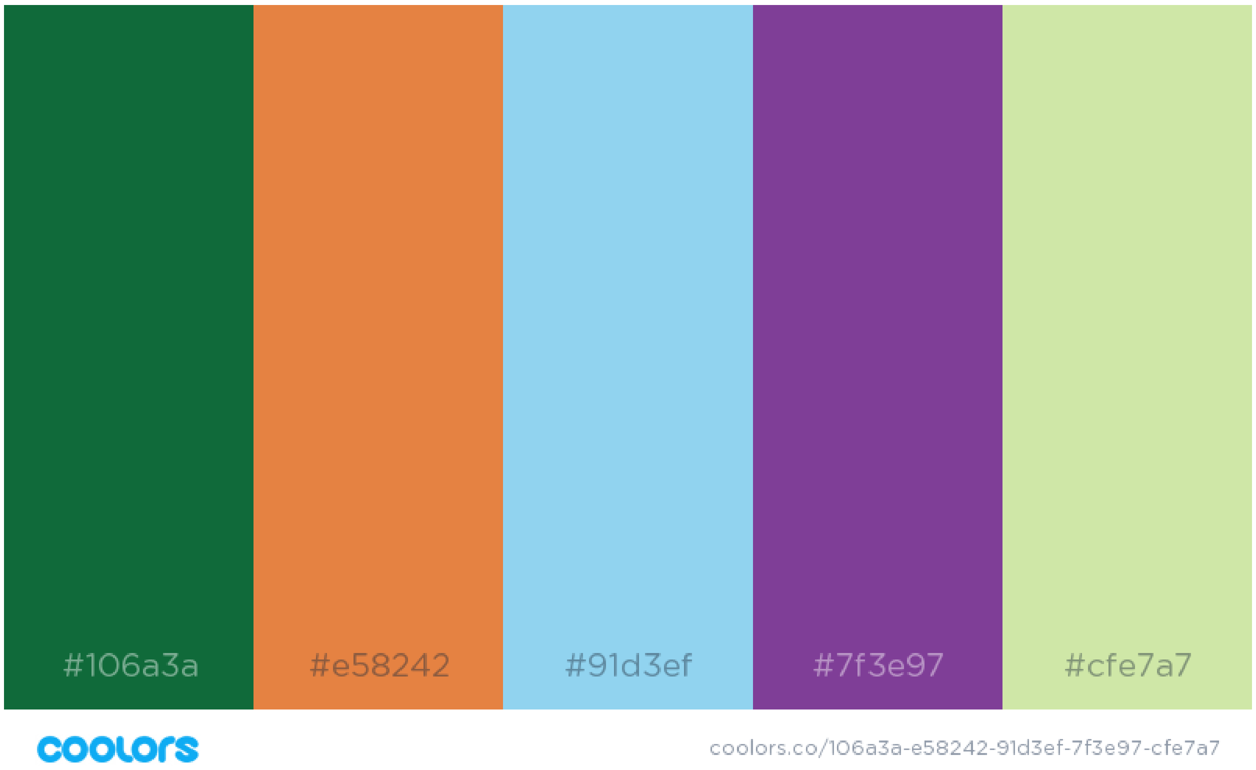


Figure W: Accentuation Pallet

- This pallet presents the colors available for specific special highlights, button colors, border colors, on selection highlight, and background colors which are just a few examples of ways that these colors can be worked in.
- These colors are supporting colors and should not be the main backdrop of any parts of the application

Product Design Specification Approval

The undersigned acknowledge they have reviewed the GOMC Capstone Product Design Specification document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Signature: ______________________________ **Date:** ____________

Print Name: ______________________________

Title: ______________________________

Role: ______________________________

Signature: ______________________________ **Date:** ____________

Print Name: ______________________________

Title: ______________________________

Role: ______________________________

Signature: ______________________________ **Date:** ____________

Print Name: ______________________________

Title: ______________________________

Role: ______________________________

Signature: ______________________ **Date:** ____________

Print Name: ______________________

Title: ______________________

Role: ______________________

Signature: ______________________ **Date:** ____________

Print Name: ______________________

Title: ______________________

Role: ______________________

Appendix A: References

The following table summarizes the documents referenced in this document.

Document Name and Version	Description	Location
Software Requirements Specification version 1.2.1	Please refer to the appendix and key terms of the listed document	Team Google Drive: https://drive.google.com/drive/folders/0B_ainqSY42cXSnl4SU5oeWg5TU0
Class diagram 1.0	A Visio file of the class diagram.	https://drive.google.com/open?id=0B8KjTyY9P8pKMENBcFVtelVITzg
GOMC manual	The manual documentation file which also has documentation of the config input fields.	gomc.eng.wayne.edu/GOMC_files/GOMC_Manual.pdf
GitHub Release Api	The api for getting information about releases of a repo.	https://developer.github.com/v3/repos/releases/