



---

## INFORMATICS 3A10

---

### Final Year Project



Group Name	The Overclockers
Group Number	25
Project Name	Eternal Vibes
Name of Project Supervisor	Brian Greaves

Name	Surname	Student Number
Ronald	Lai	201433999
David	Innes	201474149
Devandrin	Kuni	201320596
Marc	Zuze	201477488

# Table of Contents

Plagiarism Document.....	3
Gantt Chart Diagram.....	4
Resources Breakdown .....	5
Resource Breakdown Table .....	6
Projected Project Cost Estimation Table per Gantt Chart Activity .....	10
Verification and Validation Diagram.....	15
Mobile App and Website: Registration.....	16
Mobile App and Website Login.....	17
Website: Settings .....	17
Website: Ban User .....	18
Website: Ban Post.....	19
Website: Broadcast Message.....	19
Website: Statistics .....	20
Website: Logout.....	21
Mobile App: View and Change Settings.....	21
Mobile App: Newsfeed .....	22
Make a Website or Mobile App Newsfeed Post.....	22
Website and Mobile App: Like a Post .....	23
Website and Mobile App: Flag a Post.....	24
Mobile App: Radar .....	25
Mobile App: Music Artists Rating .....	26
Mobile App: Messenger.....	28
Mobile App and Website: Music Artist Profile .....	29
Mobile App and Website: Event .....	30
Appendix – Use Cases .....	31

# Plagiarism Document



## Academy of Computer Science and Software Engineering

Informatics 3 Group Project


### Plagiarism Declaration

Deliverable #	4	Team #	25	Supervisor	Brian Greaves
Team Name	The Overclockers				

1. Plagiarism is to present someone else's ideas as our own.
2. Where material created by other people has been used (either from a printed or digital source) this has been carefully acknowledged and referenced. We have used the appropriate style for citation and referencing. Every contribution to and quotation from the work of other people in this deliverable has been acknowledged through citation and reference.
3. We know that plagiarism is wrong.
  - 3.1. We understand what plagiarism is and am aware of the University's policy in this regard.
  - 3.2. We know that we would plagiarise if we do not give credit to my sources, or if we copy any part of a book, article, or Internet source without proper citation.
  - 3.3. We know that even if we only change the copied work slightly, we still plagiarise when using someone else's work without proper citation.
  - 3.4. We declare that we have created our own work throughout this deliverable and we have credited all ideas we have gained from other people's work.
4. We declare that this deliverable is our own original work.
5. We have not allowed, and will not allow, anyone to copy our work with the intention of passing it off as his or her own work.

Signature   
Full Name Marc Lermanzo Zuze  
Student # 201477488  
Date 09/05/2017  
Member 1

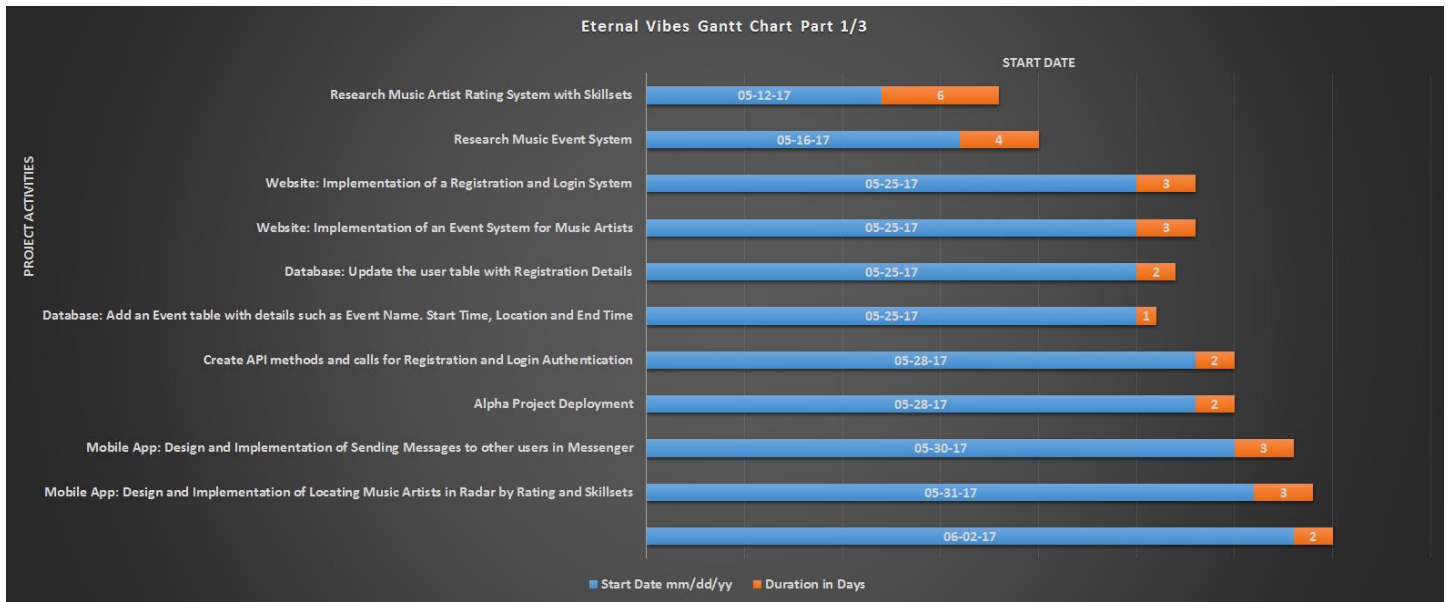
Signature   
Full Name Devandrin Kuni  
Student # 201320596  
Date 09/05/2017  
Member 2

Signature   
Full Name David Innes  
Student # 201474149  
Date 09/05/2017  
Member 3

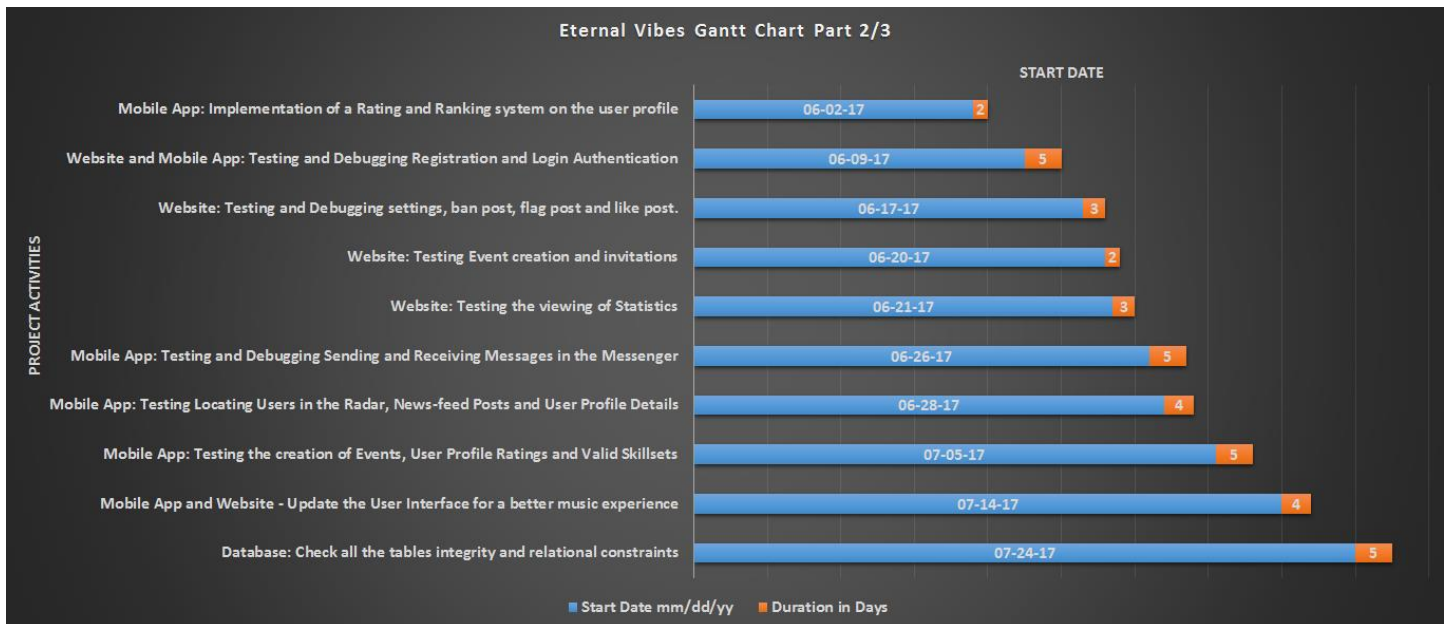
Signature   
Full Name Ronald Lai  
Student # 201433999  
Date 09/05/2017  
Member 4

# Gantt Chart Diagram

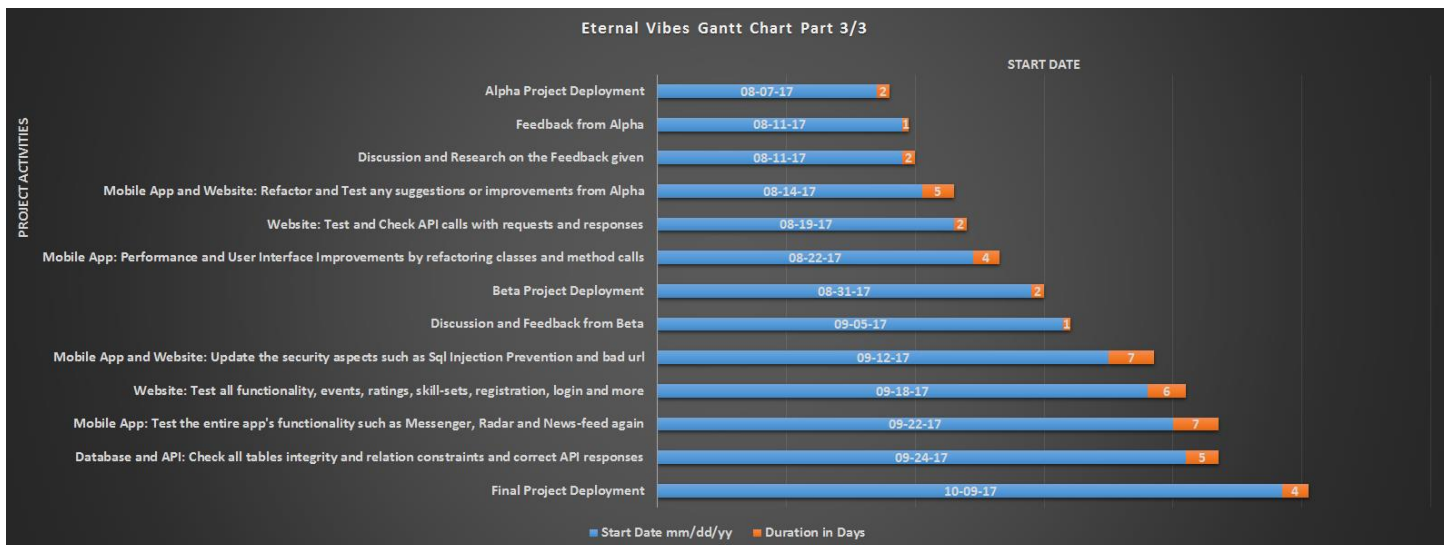
Eternal Vibes Gantt Chart Part 1/3



Eternal Vibes Gantt Chart Part 2/3



Eternal Vibes Gantt Chart Part 3/3



## Resources Breakdown

### Introduction

Our team consists of Ronald, David, Devandrin and Marc. We will all be involved in the resources breakdown and each of us will be using laptops and smartphones to create the system.

Laptop Specifications					
	Processor	RAM	Hard Drive	Brand	Other
Ronald	Intel Core i3 3.2 GHz	4GB	1TB	HP	Windows 8.1 64 Bit, Intel HD 4000 Graphics
Devandrin	Intel Core i5 3.2 GHz	8GB	1TB	HP	Windows 10 64 Bit, AMD 520 Graphics
David	Intel Pentium 2.4 GHz	4GB	1TB	Asus	Windows 10 64 Bit, Intel HD 3000 Graphics
Marc	Intel Core i3 3.0 GHz	4GB	500GB	Asus	Windows 7 Bit, Intel HD 3000 Graphics
Smartphone Specifications					
	Phone Brand Name	Processor	Memory	Camera	Storage
Ronald	Samsung Galaxy J1 Mini	1.2GHz quad-core.	768MB	5-megapixel	8GB
Devandrin	Huawei Nova	2GHz octa- core	128GB	12- megapixels	32GB
David	Motorola Nexus 6	Quad-core 2.7 GHz	3 GB	13-megapixel	32GB
Marc	Google Nexus 5	Quad-core, 2260 MHz, Krait 400	2 GB	1.3 megapixels	32 GB

## Resource Breakdown Table

Task Number	Task Name	People	Software	Hardware	Other Resources
1	Research Music Artist Rating System with Skillsets	Marc	Lucidchart, Draw.IO	Paper, pencils, pens and colours	Information from sponsors and other music artists
2	Research Music Event System	David and Marc	Lucidchart, Draw.IO	Paper, pencils, pens and colours	Information from sponsors and event organisers
3	Website: Implementation of a Registration and Login System	David	Node.js, Express, HTML and CSS	Laptop	Digital Ocean Cloud, Namecheap, Ubuntu and Njinx
4	Website: Implementation of an Event System for Music Artists	David	Node.js, Express, HTML and CSS	Laptop	Face to face communication
5	Database: Update the user table with Registration Details	David	MySQL Database	Laptop	Internet information about relational and integrity constraints
6	Database: Add an Event table with details such as Event Name. Start Time, Location and End Time	David	MySQL Database	Laptop	Internet information about relational and integrity constraints
7	Create API methods and calls for Registration and Login Authentication	David	Node.js and backend Express	Laptop	Internet for information on efficient request and response techniques
8	Alpha Project Deployment	David and Devandrin	Digital Ocean and Android	Laptop, Smartphone	Github

			Studio Device Simulator	and USB cable	
<b>9</b>	Mobile App: Design and Implementation of Sending Messages to other users in Messenger	Ronald	Android Studio and Java	Laptop	Internet information about sending and receiving messages to and from an API and Github
<b>10</b>	Mobile App: Design and Implementation of Locating Music Artists in Radar by Rating and Skillsets	Ronald and Devandrin	Android Studio and Java	Laptop	Mathematical Distance formula when given longitude and latitude co-ordinates. Github
<b>11</b>	Mobile App: Implementation of a Rating and Ranking system on the user profile	Devandrin and Marc	Android Studio and Java	Laptop, Smartphone and USB cable	Sponsor guidance on different levels of experience of a music artist. Github
<b>12</b>	Website and Mobile App: Testing and Debugging Registration and Login Authentication	Devandrin and David	Android Studio Unit Tests and Selenium Automate-d Website Testing	Laptop, Smartphone and USB cable	Internet information on different types of unit tests and lecturer guidance of efficient unit tests.
<b>13</b>	Website: Testing and Debugging settings, ban post, flag post and like post.	David	Node.js and Selenium Automate-d Website Testing	Laptop, Smartphone and USB cable	Internet information on post requirements for a ban or flag.
<b>14</b>	Website: Testing Event creation and invitations	David	Node.js and Selenium Automate-d Website Testing	Laptop, Smartphone and USB cable	Internet information on invitation styles

<b>15</b>	Website: Testing the viewing of Statistics	David	Node.js and Selenium Automate-d Website Testing	Laptop	Mathematical types of statistics, math lecturers and math notes.
<b>16</b>	Mobile App: Testing and Debugging Sending and Receiving Messages in the Messenger	Ronald	Android Unit Testing, Android Device Simulator and Java	Laptop, Smartphone and USB cable	Internet information on standard / average message lengths.
<b>17</b>	Mobile App: Testing Locating Users in the Radar, News-feed Posts and User Profile Details	Devandrin and Marc	Android Unit Testing, Android Device Simulator and Java	Laptop, Smartphone and USB cable	Basic geographical information form the internet or lecturer.
<b>18</b>	Mobile App: Testing the creation of Events, User Profile Ratings and Valid Skillsets	Devandrin	Android Unit Testing, Android Device Simulator and Java	Laptop, Smartphone and USB cable	Internet information on profile styles and layouts
<b>19</b>	Mobile App and Website - Update the User Interface for a better music experience	Devandrin	Android Unit Testing, Android Device Simulator and Java	Laptop, Smartphone and USB cable	Feedback from stakeholders and external moderators
<b>20</b>	Database: Check all the tables integrity and relational constraints	David	MySql and Selenium Automated Testing	Laptop	Internet example of good integrity constraints and how to implement them
<b>21</b>	Alpha Project Deployment	David and Devandrin	Digital Ocean and Android Simulator, MS PowerPoint slides	Laptop	UJ Wifi, Namecheap, njinx and Github.



<b>22</b>	Feedback from Alpha	Mentors and External Moderators	Google Chrome Browser	Laptop, Paper, Pens and Pencils, Pointers	Feedback from stakeholders and other music artists.
<b>23</b>	Discussion and Research on the Feedback given	Marc, Devandrin, David and Ronald	Gmail and Whatsapp	Paper, Pens and Pencils	Discussing the changes with stakeholders.
<b>24</b>	Mobile App and Website: Refactor and Test any suggestions or improvements from Alpha	Marc, Devandrin, David and Ronald	Android Studio, Node.js, HTML5 and Express	Laptop, Paper, Pens and Pencils, and Smartphones	Internet information about refactoring a website or mobile app.
<b>25</b>	Website: Test and Check API calls with requests and responses	David	Node.js, Express and Selenium Automated Testing	Laptops	Internet information about mobile app interaction with Node.js
<b>26</b>	Mobile App: Performance and User Interface Improvements by refactoring classes and method calls	Ronald and Devandrin	Android Studio, Java and Android Device Simulator	Laptop and Smartphone	Computer Science information from peers and lecturers on efficient data structures for complex data processing and temporary storage.
<b>27</b>	Beta Project Deployment	David and Devandrin	Digital Ocean and Android Simulator, MS PowerPoint slides	Laptop	Namecheap, njinx and Github.
<b>28</b>	Discussion and Feedback from Beta	Marc, Devandrin, David and Ronald	Email	Laptop, Paper and Pens	Face to face communication
<b>29</b>	Mobile App and Website: Update the security	David and Devandrin	Sql-injection (an express module that	Laptop	Internet information and lecturer advice

	aspects such as Sql Injection Prevention and bad url		prevents sql injections)		on SQL-Injection prevention and Internet connection
<b>30</b>	Website: Test all functionality, events, ratings, skill-sets, registration, login and more	David	Selenium, Node.js and Express	Laptop	Letting other users test the system
<b>31</b>	Mobile App: Test the entire app's functionality such as Messenger, Radar and News-feed again	Devandrin, Ronald and Marc	Android Studio, Android Device Simulator and Java	Laptop, Smartphone and USB cable	Letting other users test and use the system and Android Unit Testing
<b>32</b>	Database and API: Check all tables integrity and relation constraints and correct API responses	David and Marc	MySQL, Selenium, Node.js and Express	Laptop	Internet connection
<b>33</b>	Final Project Deployment	David, Devandrin, Ronald and Marc	Digital Ocean and Android Simulator, MS PowerPoint slides	Laptops, LCD screens, Smartphones, and Routers	UJ-Wifi, smartphone wi-fi

### Projected Project Cost Estimation Table per Gantt Chart Activity

Activity Name	Cost Per Hour	Cost Per Day (8 Hours)	Number of Days	Cost Per Activity
Website: Implementation of a Registration	R400	$R400 \times 8 = R3200$	3	$R3200 \times 3 = R9600$

and Login System				
Website: Implementation of an Event System for Music Artists	R400	$R400 \times 8 = R3200$	3	$R3200 \times 3 = R9600$
Database: Update the user table with Registration Details	R400	$R400 \times 8 = R3200$	1	$R3200 \times 1 = R3200$
Database: Add an Event table with details such as Event Name, Start Time, Location and End Time	R350	$R300 \times 8 = R2400$	2	$R2400 \times 2 = R4800$
Create API methods and calls for Registration and Login Authentication	R300	$R300 \times 8 = R2400$	2	$R2400 \times 2 = R4800$
Alpha Project Deployment	R300	$R300 \times 8 = R2400$	4	$R2400 \times 4 = R9600$
Mobile App: Design and Implementation of Sending Messages to other users in Messenger	R450	$R450 \times 8 = R3600$	6	$R3600 \times 6 = R21600$
Mobile App: Design and Implementation				

of Locating Music Artists in Radar by Rating and Skillsets	R450	$R450 \times 8 = R3600$	9	$R3600 \times 9 = R32400$
Mobile App: Implementation of a Rating and Ranking system on the user profile	R450	$R450 \times 8 = R3600$	4	$R3600 \times 4 = R14400$
Website and Mobile App: Testing and Debugging Registration and Login Authentication	R400	$R400 \times 8 = R3200$	5	$R3200 \times 5 = R16000$
Website: Testing and Debugging settings, ban post, flag post and like post.	R400	$R400 \times 8 = R3200$	5	$R3200 \times 5 = R16000$
Website: Testing Event creation and invitations	R400	$R400 \times 8 = R3200$	5	$R3200 \times 5 = R16000$
Website: Testing the viewing of Statistics	R400	$R400 \times 8 = R3200$	3	$R3200 \times 3 = R9600$
Mobile App: Testing and Debugging Sending and Receiving Messages in the Messenger	R450	$R450 \times 8 = R3600$	6	$R3600 \times 6 = R21600$

Mobile App: Testing Locating Users in the Radar, News- feed Posts and User Profile Details	R450	$R450 \times 8 = R3600$	7	$R3600 \times 7 = R25200$
Mobile App: Testing the creation of Events, User Profile Ratings and Valid Skillsets	R450	$R450 \times 8 = R3600$	9	$R3600 \times 9 = R32400$
Mobile App and Website - Update the User Interface for a better music experience	R450	$R450 \times 8 = R3600$	5	$R3600 \times 5 = R18000$
Database: Check all the tables integrity and relational constraints	R350	$R350 \times 8 = R2800$	5	$R2800 \times 5 = R14000$
Mobile App and Website: Refactor and Test any suggestions or improvements from Alpha	R550	$R550 \times 8 = R4400$	6	$R4400 \times 6 = R26400$
Website: Test and Check API calls with requests and responses	R500	$R500 \times 8 = R4000$	6	$R4000 \times 6 = R24000$
Mobile App: Performance				

and User Interface Improvements by refactoring classes and method calls	R350	$R350 \times 8 = R2800$	6	$R2800 \times 5 = R16800$
Beta Project Deployment	R500	$R500 \times 8 = R4000$	5	$R4000 \times 5 = R20000$
Mobile App and Website: Update the security aspects such as Sql Injection Prevention and bad url	R450	$R450 \times 8 = R3600$	7	$R3600 \times 7 = R25200$
Website: Test all functionality, events, ratings, skill-sets, registration, login and more	R400	$R400 \times 8 = R3200$	6	$R3200 \times 6 = R19200$
Mobile App: Test the entire app's functionality such as Messenger, Radar and News-feed again	R450	$R450 \times 8 = R3600$	7	$R3600 \times 7 = R25200$
Database and API: Check all tables integrity and relation constraints and correct API responses	R350	$R350 \times 8 = R2800$	5	$R2800 \times 5 = R14000$

Final Project Deployment	R600	$R600 \times 8 = R4800$	4	$R4800 \times 4 = R19200$
--------------------------	------	-------------------------	---	---------------------------

**Total Cost for All Activities:**  $R9600 + R9600 + R3200 + R4800 + R4800 + R9600 + R21600 + R32400 + R14400 + R48000 + R9600 + R21600 + R25200 + R32400 + R18000 + R24000 + R26400 + R24000 + R16800 + R20000 + R25200 + R19200 + R25200 + R14000 + R19200 = R478800$

**Transportation Costs:**  $R30 \text{ per day} \times 260 \text{ work days} \times 4 \text{ members} = R31200$

**Electricity Costs:**  $R3000 \text{ per month} \times 12 \text{ months} \times 4 \text{ members} = R144000$

**Digital Ocean Services:**  $R130 \text{ per month} \times 12 = R1560$

**Project Grand total:**  $R478800 + R31200 + R144000 + R1560 = \mathbf{R655560}$

## Verification and Validation Diagram

### Introduction

David will be using Selenium Webdriver because it allows for QA Test automation, the tests will be written in java. It will allow for tests to be run in the development environment to test all the page's elements to ensure that new updates don't disrupt original functionality. Marc will be manually testing the website's styling features such as scrolling, animations and colour themes.

For the mobile app, Android has built in unit testing which will be used to test various features of the app. Devandrin and Ronald will be using these unit tests to simulate user interaction such as sending messages and making posts. Registration, Login and locating music artists will be tested manually as well as any new UI features that need to be added to the mobile app. This is because automated tests do not know the look and feel of software when it is shown to a user.

## Mobile App and Website: Registration

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Verification of Website and Mobile App Registration Details	Email, Password, Alias, Firstname, Lastname, DateofBirth and Preferred Genre	All input fields have correct syntax and contain no errors.	Register the user by sending the details to the database. Redirect user to login page.
2	Verification of Website and Mobile App Registration Details	Email entered	Incorrect syntax	Alert user to enter a valid email address
		Email entered	Already exists	Display message "Cannot register again message"
		Password entered	Incorrect syntax	Prompt user to check password
			Empty Field, Prompt user to enter a password	Password is not empty; a tick is shown next to the password field.
		Password Confirm	Empty Field	Prompt to enter password to match
			Does not match 1 <sup>st</sup> password	Prompt to re-enter password
		Alias entered	Empty	Alert above field box to enter an alias.
		Alias entered	Alias exists	Alert user to enter a different alias.
		First name entered	Empty	Alert above field box



		Last name entered	Empty	Alert above field box
		Date of birth entered	Empty	Alert above field box to select a date
		Preferred Genre	Not selected	Alert User to select a genre

### Mobile App and Website Login

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Mobile App and Website Login Authentication	Email Password	User exists, and password is correct	Authenticate user and log them in, continue to website dashboard or newsfeed in the mobile app.
2		Email Password	User exists but password is incorrect.	Display incorrect password authentication message
3		Email Password	User does not exist	Display not registered message
4		Email Password	Server or database connection error	Display connection error message

### Website: Settings

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Updating Settings	User changes various settings	No internet, Server or Database connection errors. User is logged in.	Update the user's settings

2		User changes various settings	Server and / or database connection not established.	Display message "Connection Error. Cannot update and change settings"
3		User changes various settings	No internet, Server or Database connection errors. User is not logged in.	The settings will not be loaded on the webpage. Display message "You need to be logged in to view settings."

### Website: Ban User

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Ban User Verification	Administrator clicks a user profile and bans them.	Database and Server connections established, Administrator is logged in.	Ban the selected user and load the next user to ban (If they are flagged enough times.) The banned user cannot send messages, make posts or locate other users.
2		Administrator clicks a user profile and bans them.	Database and Server connections established, but the Administrator is not logged in.	Display message "A logged in Administrator is required to ban the selected user."
3		Administrator clicks a user profile and bans them.	No Database and Server connections could be established	Display message "Connection Error, please try again later."

### Website: Ban Post

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Ban Post Validation	Administrator clicks a user profile and bans a post or a collection of posts	Database and Server connections established, Administrator is logged in. The selected user has at least one post.	The selected user's post is now banned. The user will now receive a notification of their banned post.
2		Administrator clicks a user profile and bans a post or a collection of posts	Database and Server connections established, Administrator is logged in. The selected user has no posts.	Display message "Select a user with at least one post to ban it."
3		Administrator clicks a user profile and bans a post or a collection of posts	Database and Server connections established, user has at least one post but the Administrator is not logged in.	Display message "A logged in Administrator is required to ban the selected user's post."
4		Administrator clicks a user profile and bans a post or a collection of posts.	No Database and Server connections could be established.	Display message "Connection Error, please try again later."

### Website: Broadcast Message

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Broadcast Message Confirmation	Administrator enters a message	Database and Server connections established,	Display the message to all users.

		to broadcast to all the users.	Administrator is logged in, a message has been entered the input field.	
2		Administrator enters a message to broadcast to all the users.	Database and Server connections established, Administrator is logged in, nothing input field	Display a notification to the Administrator to input a message to send to all the users.
3		Administrator enters a message to broadcast to all the users.	Database and Server connections established, Administrator is not logged in.	Do not load the Broadcast Page. It is only accessible by logged in Administrators.
4		Administrator enters a message to broadcast to all the users.	No Database and Server connections established	Display message "Connection Error, please try again later."

### Website: Statistics

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Display Correct Statistics	User views various statistics about music artists	Database and Server connections established. All relevant data is stored in the database. User is logged in.	Calculate, generate and display the statistics to the user.
2		User views various statistics about music artists	Database and Server connections established. All relevant data is stored in the database but user is not logged in.	Display message "You must be logged in to view statistics."

3		User views various statistics about music artists	Database and Server connections established. User is logged in but no data is found or stored about the selected statistic(s)	Display message "No statistics can be found for this category or music artist."
---	--	---	---	---

#### Website: Logout

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Correct Logout Execution	Logout button is clicked	No Database or Server connection errors. No internal errors.	Prompt to logout, end the session, the cookie is removed, user is now redirected to the landing page.
2		Logout button is clicked	Unexpected error	Display message "Cannot logout, please try again later."

#### Mobile App: View and Change Settings

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Discovery Settings Detection	Discovery is turned off or on	Connection between the server and database is working	Update the user's settings
2		Discovery is turned off or on	No connection to the server or database.	Display message "Cannot update settings"
1	Search Radius Verification	Search Radius is changed	Connection between the server and database is working.	Update the user's search radius

2		Search Radius is changed	No connection to the server or database.	Display message "Cannot update radius"
1	Correct Logout Execution	Logout button is clicked	No errors	Prompt to logout, all data cleared, application will start from login
2		Logout button is clicked	Unexpected error	Display cannot logout message

### Mobile App: Newsfeed

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Newsfeed functionality	View the newsfeed	Database or server connection has been established,	The user will be able to view the newsfeed list and view posts from other music artists.
2		View the newsfeed	No database or server connection.	Display message "Connection Error please try again later." The newsfeed will not be loaded.

### Make a Website or Mobile App Newsfeed Post

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Newsfeed post validation	Make a new post	Database or server connection is established	The post is sent to the server and stored in the database. The new post is now shown in the newsfeed.

2		Make a new post	Database or server connection cannot be established	Notify user that the posts cannot be made due to a database / server connection error.
3		Make a new post	Make an empty post	Alert the user to enter something to post.

### Website and Mobile App: Like a Post

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Validate Posts	Like Post Button Clicked	Database and Server connections established. Post does exist,	Notify the user that the selected post has been liked.
2		Like Post Button Clicked	Database and Server connections established. Post does not exist,	Notify the user to like an existing post
3		Like Post Button Clicked	No Database or Server connection established.	Display message "Connection Error, cannot like the post."
4		Like Post Button Clicked Twice	Database and Server connections established. Post already exists	Notify the user that selected post has been disliked.
5		Like Post Button Clicked Twice	Database and Server connections established. Post does not exist,	Notify the user to unlike an existing post

6		Like Post Button Clicked Twice	No Database or Server connection established.	Display message "Connection Error, cannot unlike the post."
---	--	--------------------------------	---	---

### Website and Mobile App: Flag a Post

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Flag Post Validation	Flag Post Button Clicked	Database and Server connections established. Post exists.	Notify the user that selected post has been flagged.
2		Flag Post Button Clicked	Database and Server connections established but post does not exist.	Notify the user to flag an existing post
3		Flag Post Button Clicked	No Database or Server connection established.	Display message "Connection Error, cannot flag the post."
4		Flag Post Button Clicked Twice	Database and Server connections established. Flagged post exists.	Notify the user that the selected post has been unflagged.
5		Flag Post Button Clicked Twice	Database and Server connections established No post exists.	Notify the user to unflag an existing flagged post.
6		Flag Post Button Clicked Twice	No Database or Server connection established.	Display message "Connection Error, cannot flag the post."



## Mobile App: Radar

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Locating Music Artist Validation	Locate and View All Music Artists within a search radius (in Km) set by the user (regardless of any other specifics)	All database and server connections are established, at least one user is within the specified search radius.	Display a list of located music artists that shows all details such as username, skillset and rankings.
2		Locate and View All Music Artists within a search radius (in Km) set by the user (regardless of any other specifics)	All database and server connections are established, but no users were found within the search radius.	Display message "No music artists could be found within the specified search radius. Please try a different search radius."
3		Locate and View All Music Artists within a search radius (in Km) set by the user (regardless of any other specifics)	The server and / or database connection is not established	Display a connection error message.
1	Music Artist Skillset Validation (Skillsets include vocals, pianist, drummer and anything physically related to creation of music.)	Search and view music artists by a collection of skillsets within a search radius.	Database and Server connections established, located within a radius and music artists have at least one skillset.	A list of located music artists based on a selected skillset with their details are displayed in a list.
2		Search and view music artists by a specified skillset within a search radius.	Database and Server connections established, located within a radius but the music artists have no skillsets.	Display message that the searched users have no skillsets yet.

3		Search and view music artists by a specified skillset within a search radius.	Database and Server connections established, located within a radius, the music artists have a skillset but does not match the specified one the user is searching for.	Display message that no users of the specified skillset exists.
4		Search and view music artists by a specified skillset within a search radius	Database and Server connections established but no users located within a radius	Display message "Cannot find any music artists within the search radius."
5		Search and view music artists by a specified skillset within a search radius.	No Database or Server connection could be established.	Display message "Connection Error, please search again later."

#### Mobile App: Music Artists Rating

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Music Artists Sorted Rating Verification	Search and view music artists by highest or lowest Ratings, within a search radius specified by the user.	Database and Server connections established, users located within search radius, the located music artist(s) have at least one rating given by another music artist.	A list of located music artists based on highest or lowest ratings with their details are displayed.
2		Search and view music artists by highest or lowest Ratings, within a search radius	Database and Server connections established, users located within search radius, but the located music	Display message "No ratings have been given to the located user(s)"

		specified by the user.	artists have received no ratings yet,	
<b>3</b>		Search and view music artists by highest or lowest Ratings, within a search radius specified by the user.	Database and Server connections established, but no users found within the search radius.	Display message "Cannot display users with highest / lowest Ratings because no users were found within the specified search radius"
<b>4</b>		Search and view music artists by highest or lowest Ratings, within a search radius specified by the user	Database and / or server connection failed.	Display a connection error message
<b>1</b>	Music Artists Specified Rating Verification	Search and view Located music artists by a specific rating	Database and Server connections established, users located within search radius, the music artist(s) meet that specific rating chosen by the user	Display a list containing the details of the user(s) that meet that specific rating.
<b>2</b>		View Located Music Artists by a specific rating	Database and Server connections established, users located within search radius but the music artist(s) do not meet that specific rating or have not received a rating yet.	Display message "Cannot find users with the specified rating."
<b>3</b>		View Located Music Artists by a specific rating,	Database and Server connections established, but no	Display message "Cannot display users with the

		within a search radius	users found within the search radius.	specified ratings because no users were found within the specified search radius”
4		View Located Music Artists by a specific rating, within a search radius	Database and / or server connection failed.	Display a connection error message

### Mobile App: Messenger

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Correct Displaying of Messenger Contacts last sent messages.	View a list of contacts and the last message sent or received.	Database and Server connection established, relevant user account is not banned.	View a list of contacts that each have a last message received by the user or sent by the contact.
2		View a list of contacts and the last message sent or received.	Database and Server connection established, but relevant user(s) account are banned.	Display notification that no messages can be sent to this user until he/she is unbanned.
3		View a list of contacts and the last message sent or received.	The Server and / or Database connection cannot be established.	Display message “Internal Connection Error. Cannot send or receive the latest messages.”
1	Verification of sending a message to a selected contact.	Send a message(s) in the Messenger to a selected contact	Database and server connections are established, the message length < 250 characters, selected user is not banned.	Send the message to the selected contact.  View the sent message for that contact.

2		Send a message(s) in the Messenger	Database and server connections are established, Length of the message > 250 characters, selected user is not banned.	Notify the user to shorten the message before sending it.
3			The message length < 250 characters, database and server connections are established, but selected user is banned.	Display message "Your message cannot be sent to this banned user. Please send a message to a user that is not banned."

### Mobile App and Website: Music Artist Profile

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Profile Viewing Correctness	User views his/her own profile.	Database and Server connections established. User is not banned.	The latest user profile details are displayed.
2		User views his/her own profile.	Database and Server connections not established.	Display message "Connection Error. Cannot load user profile details."
3		User views his/her own profile.	Database and Server connections established. User is banned.	Cannot view any details.  Display message "Your profile has been banned, please contact an administrator."
1	Editing Profile Details Verification	User edits various details such as preferred genre or	Database and Server connections	Update the user's profile details and display it.

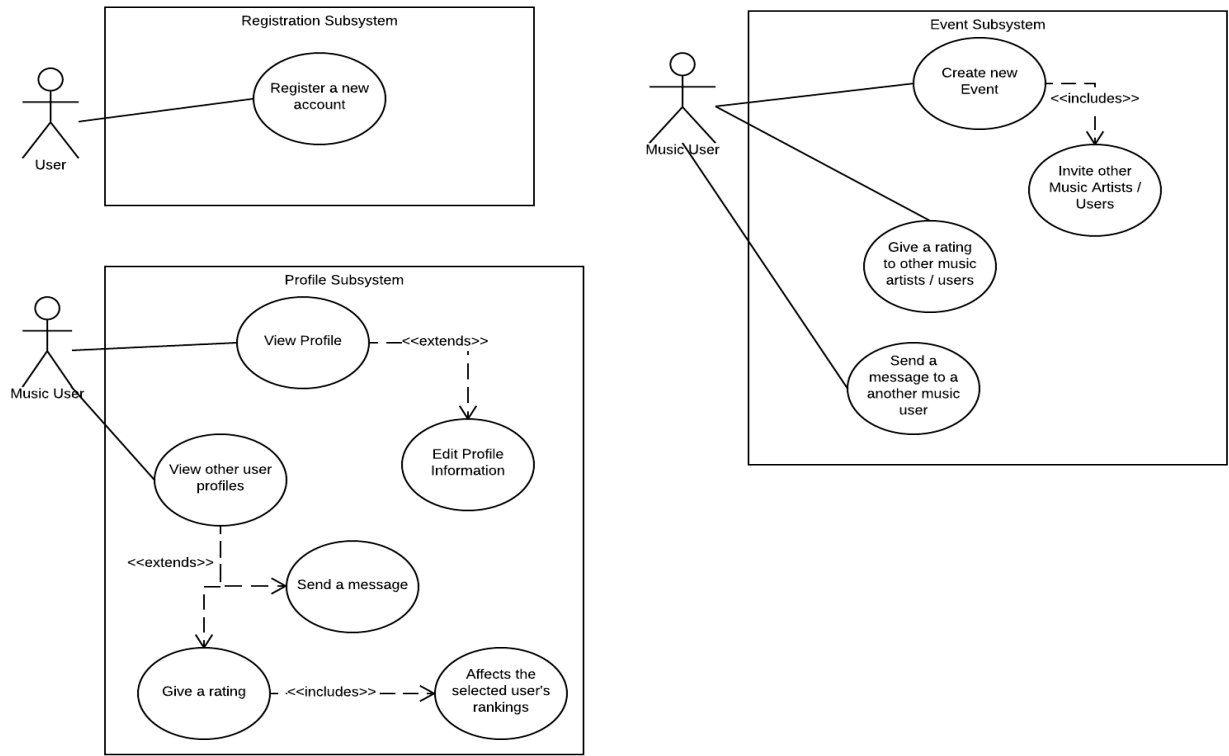
		adds a new skillset.	established, User is not banned.	
2		User edits various details such as preferred genre or adds a new skillset.	Database and Server connections established but the user is banned.	Display message that the user must unban their account to change profile details.
3		User edits various details such as preferred genre or adds a new skillset.	Database and Server connections not established.	Display message "Connection Error. Could not update your profile details."

### Mobile App and Website: Event

#	<u>Test Name</u>	<u>Input</u>	<u>Test Case / Condition</u>	<u>Output</u>
1	Event creation validation	User creates a new music event.	Database and Server connections established.	The event is created and the details such as name and location are displayed.
2		User creates a new music event.	Database and Server connections not established.	Display message "Connection Error, could not create an event. Please try again later."
1	Event Invitation and Acceptance verification	The user sends an invitation to a selected contact or a group of contacts.	Database and Server connections established. At least one other contact accepts the invitation.	The invitation is sent to the selected contact(s) and the event is created.
2		The user sends an invitation to a selected contact or a group of contacts.	Database and Server connections not established.	Display message "Connection Error. Could not sent invitation."

3		The user sends an invitation to a selected contact or a group of contacts.	Database and Server connections established, no contact accepts the invitation.	Display message “No one accepted the invitations, the event was not created.”

### Appendix – Use Cases



#### Registration Subsystem

1a.) The user can register a new account on EternalVibes.me by providing their details.

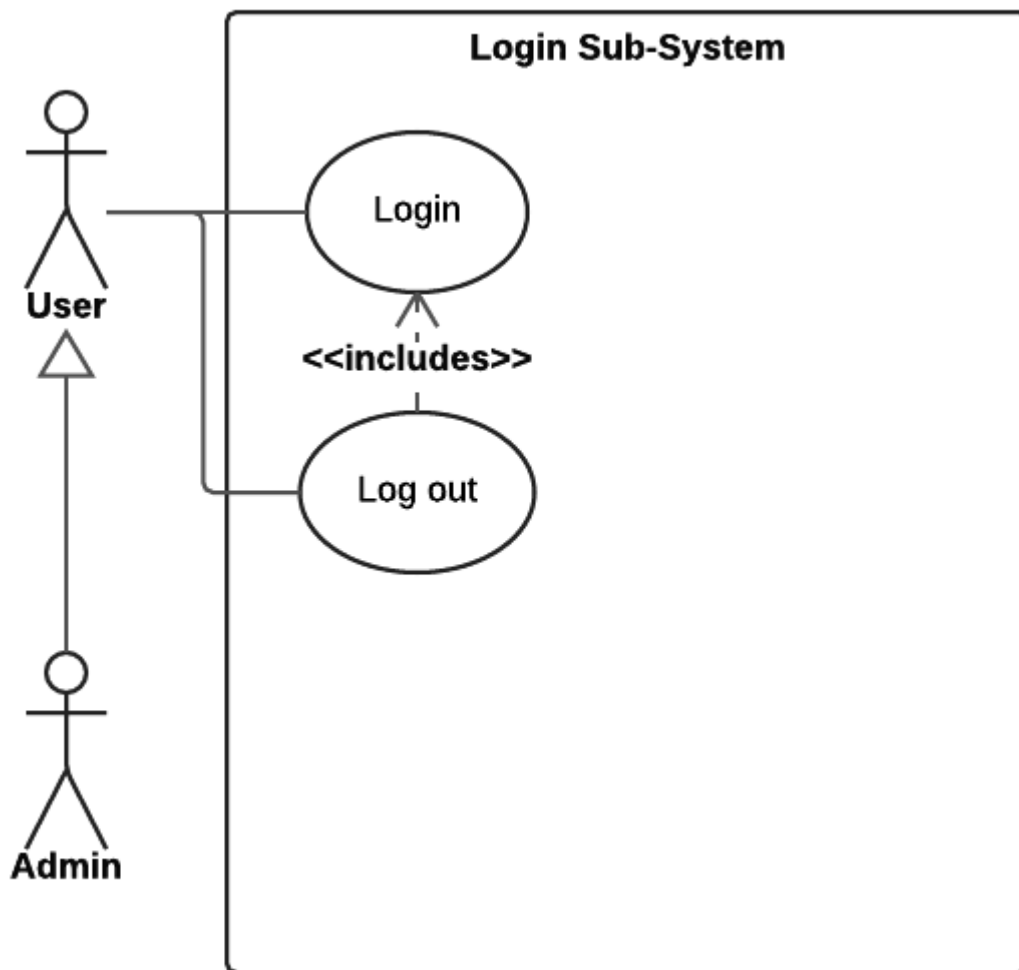
#### Event Subsystem

2a.) A registered and logged in user can create a new event for other artists to join and help out with some work based on a skillset such as vocals or drummer.

Profile Subsystem.

3a.) A logged in user can view their profile and make edits to their personal information.

b.) They can also view other profiles of located music artists; they can view their ratings, give them a rating and also message them through the mobile app's messenger.



Use Case: Login

1. Users/Admin Enters Login Credentials
2. User i/Admin Logged into System



3. System is redirected to User is/Admin dashboard

#### Extensions

##### 1a) User's/Admin's Login credentials are incorrect

1. System notifies User/Admin
2. User/admin re-enters credentials

##### 1b) User/Admin account does not exist

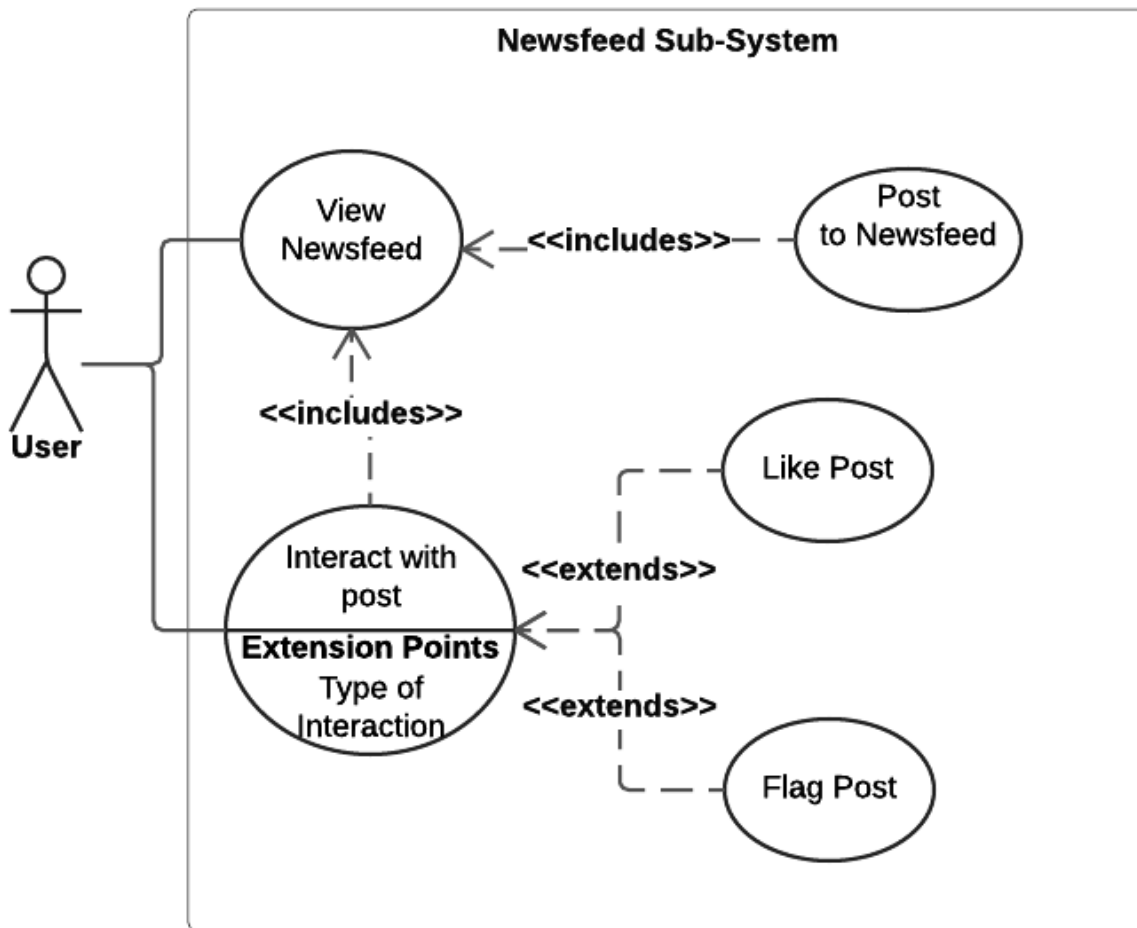
1. Notify User/Admin
2. User/Admin goes to registration page

##### 1c) User/Admin forgets login credentials

1. User/Admin enters email address linked to account
2. System sends an email to User/Admin with their Login credentials
3. User/Admin tries again

#### Use Case: Logout

1. User/Admin is finished using the system
2. User Logs out of System
3. System redirects User/Admin to login/registration page
4. User closes system



#### Use Case: View Newsfeed

1. User goes to Newsfeed
2. System loads Newsfeed
3. User browses Newsfeed

#### Extension

##### 2a) No internet connection

1. System waits for internet connection
2. System loads newsfeed

#### Use case: Post to Newsfeed

1. User goes to Newsfeed
2. System displays Newsfeed
3. User browses Newsfeed
4. User adds a post
5. System uploads post

6. User's Newsfeed is updated with new posts

#### Extension

1a) User discards post

1. System discards post

5a) No internet connection

1. Post not uploaded
2. System waits for internet connection
3. System uploads post

#### Use Case; Interact with post

1. User browses Newsfeed
2. User views an Artist's post
3. User Interacts with post

#### Use Case: Like Post

1. User browses Newsfeed
2. User views an Artist's post
3. User Interacts with post
4. User Likes/Flags post

#### Extension

4a) User Likes Post

1. System notifies Artist that their post has been liked

4b) User Flags Post

1. System notifies Admin on flagged post

1. User chooses Artist to message
2. System displays chat between the Users
3. User types a message
4. User sends message
5. System notifies Artist that new message received

#### Extensions

### 3a) Message discarded

1. User leaves chat
2. System discards message

### 4a) No internet connection

1. System notifies user
2. System waits for internet connection
3. System sends message when connection is established

### Use case : Receive a message

1. Server sends a notification to User
2. User opens chat with Artist
3. User reads received message
4. User responds to message

### Extension

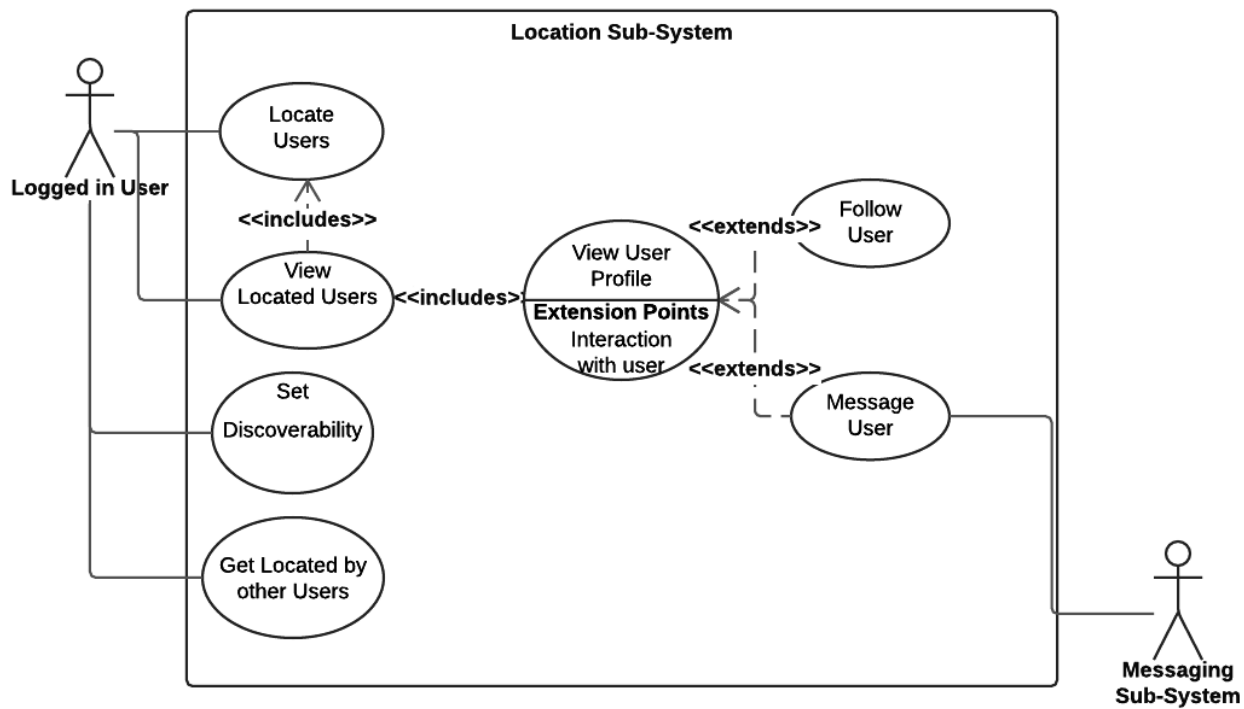
#### 1a) No Internet connection

1. System waits for a connection
2. System sends notification once connection is established

#### 4) User does not respond message

### Use Case: Notify User

1. System notifies User that a message has been received
2. System notifies user based on internet connection



#### Use Case: Set discoverability

1. User sets discoverability on or off
2. System updates User's discoverability status
3. System stores User's location

#### Extension

##### 2a) No internet connection established

1. System waits for internet connection
2. System updates User's discoverability status once internet connection established

#### Use Case: Locate User

1. User sets discoverability on
2. System updates User's discoverability status
3. System stores User's location
4. System locates nearby Artists
5. System displays located Artists
6. User views located Artists

#### Use Case: View User Profile

1. System locates nearby Artists
2. System displays located Artists
3. User views located Artists
4. User views a Artist's profile

#### Use Case: Follow User

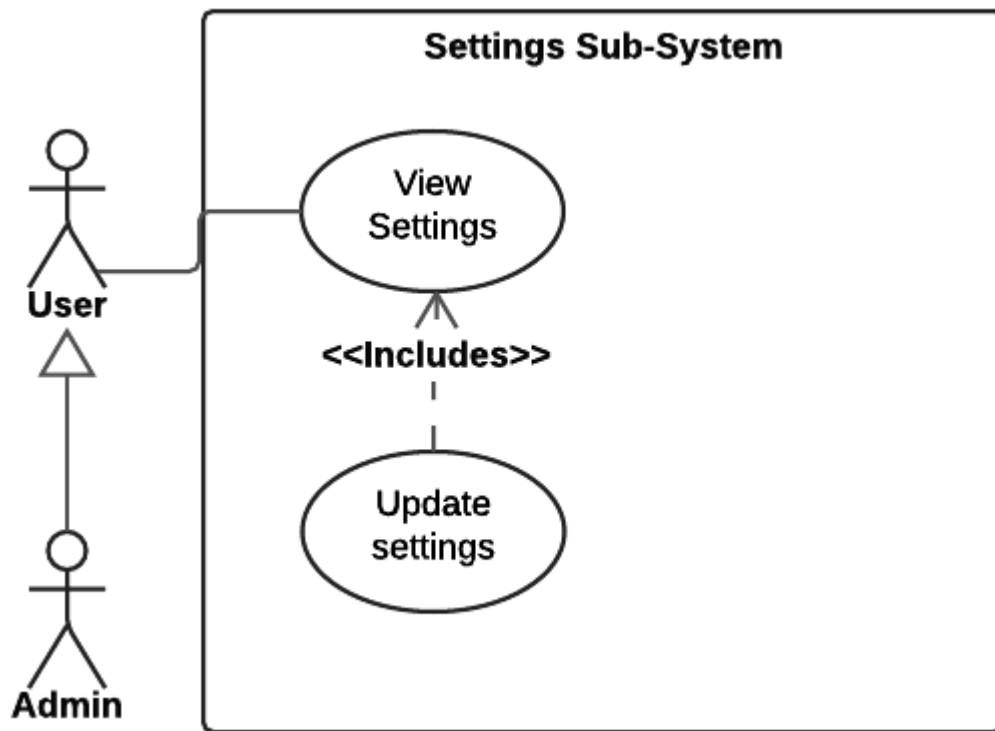
1. System locates nearby Artists
2. System displays located Artists
3. User views located Artists
4. User views a Artist's profile
5. User Follows Artist
6. System updates User's following
7. System updates Artist's Followers
8. System updates User's newsfeed
9. System notifies Artist

#### Use Case: Message User

1. System locates nearby Artists
2. System locates nearby Artists
3. System displays located Artists
4. User views located Artists
5. User views a Artist's profile
6. User sends message to Artist
7. System notifies Artist
8. System stores conversation

#### Use Case; Get Located by other Users

1. User's discovery is set on
2. System updates User's discoverability status
3. System stores User's location
4. Artists locate User
5. Artists interact with user
6. System notifies User

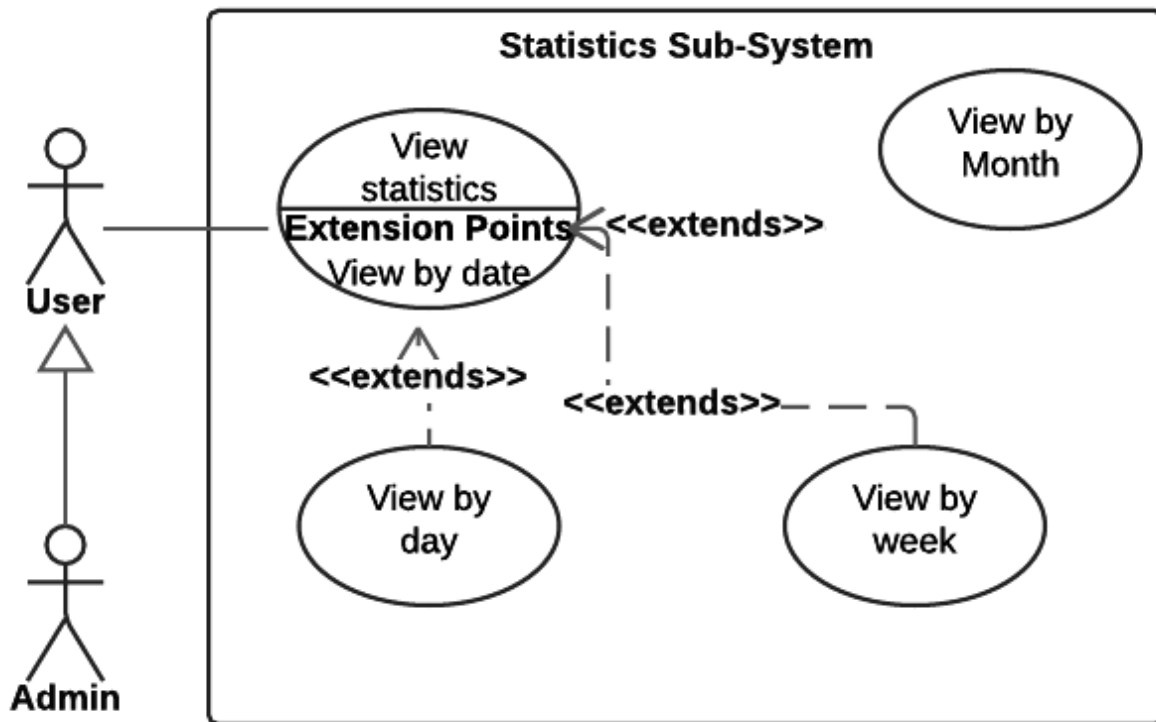


Use Case: View settings

1. User opens settings
2. System displays settings

Use Case: Update settings

1. User opens settings
2. System displays settings
3. System validates settings
4. System saves settings



Use Case: View statistics

1. System generates statistics
2. User/Admin views statistics

Use case: View by day

1. System generates statistics
2. User/Admin views statistics
3. User /Admin requests statistics view by day
4. System displays statistics by day

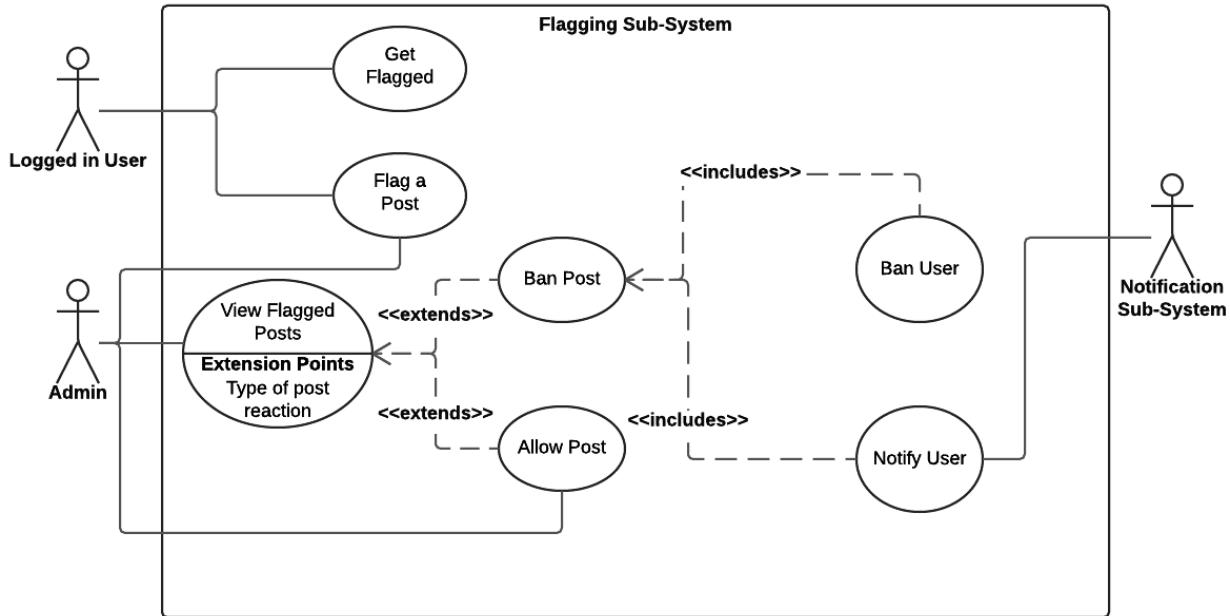
Use case: View by week

1. System generates statistics
2. User/Admin views statistics
3. User /Admin requests statistics view by week
4. System displays statistics by week

Use case: View by month



1. System generates statistics
2. User/Admin views statistics
3. User /Admin requests statistics view by month
4. System displays statistics by month



#### Use Case: Flag a post

1. User/Admin view an Artist post
2. User/Admin Flags post
3. System updates flagged posts
4. System updates how many times post has been flagged
5. System notifies Admin

#### Extension

##### 2a) Cancel post flag

1. System updates flagged posts
2. System updates how many times post has been flagged

#### Use Case: Get Flagged

1. System flags User's post based on other Artist's opinion
2. System stores post as flagged

### 3. System notifies Admin

Use case: View flagged Posts

1. Admin selects flagged posts
2. System displays all flagged posts

Extensions

2. No flagged posts available

Use Case: Ban Post

1. Admin selects flagged posts
2. System displays all flagged posts
3. Admin Bans Post
4. System Removes Post from User's newsfeed and Artist's who follow User
5. System notifies user
6. System updates flagged posts

Extension

3a) Admin cancels ban

1. System keeps post as flagged

Use Case: Ban User

1. Admin selects flagged posts
2. System displays all flagged posts
3. Admin Bans Post
4. System Removes Post from User's newsfeed and Artists who follow User
5. System notifies user
6. System updates flagged posts
7. Admin Bans User
8. System notifies user on Ban

Extensions

6a) Admin cancels ban

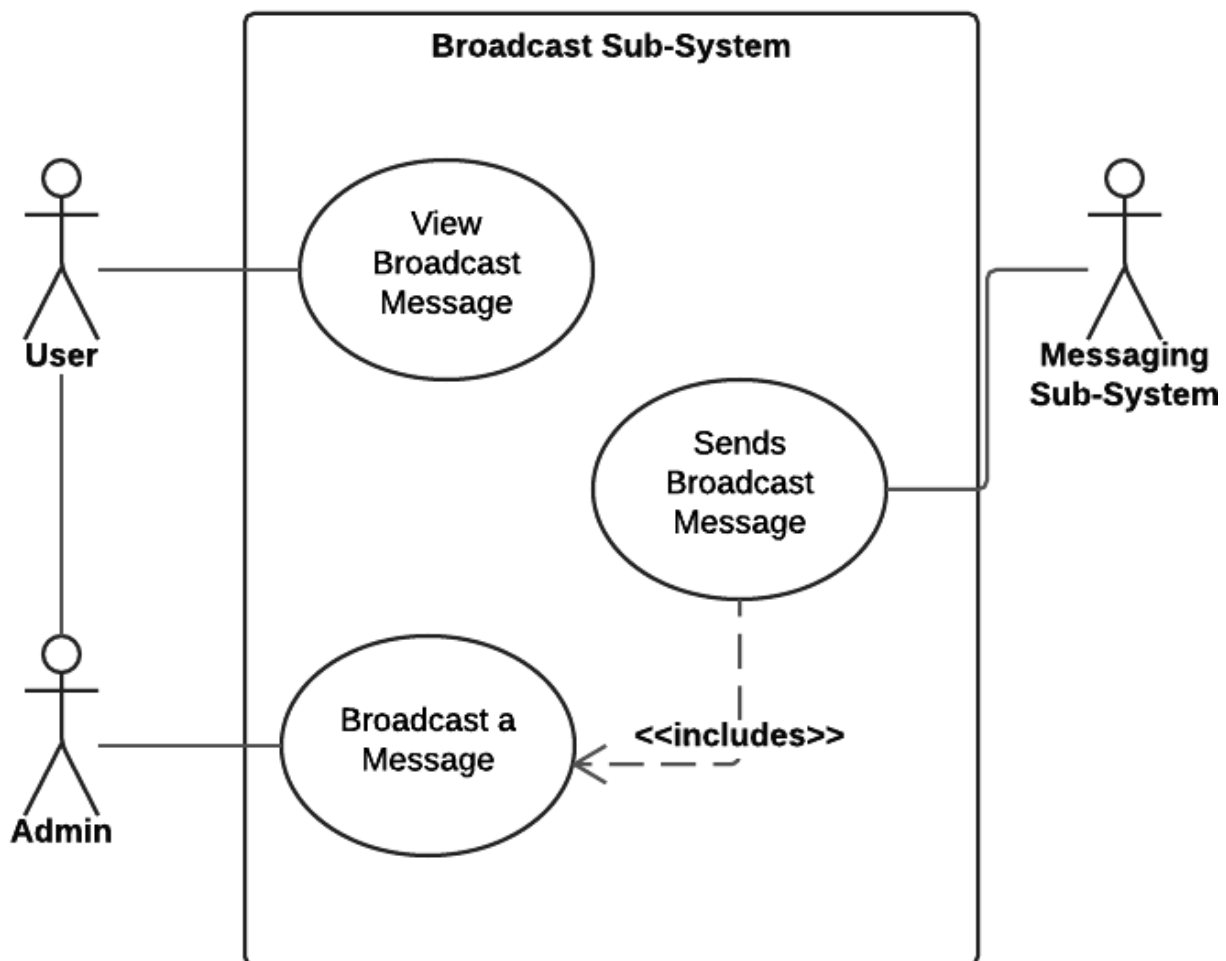
#### Use Case: Allow Post

1. Admin selects flagged posts
2. System displays all flagged posts
3. Admin approves post and removes flag
4. System keeps post on User's newsfeed and Artists who follow User
5. System updates flagged posts

#### Extensions

##### 3a) Admin cancels approval

1. System keeps post as flagged



#### Use Case: View Broadcast Message

1. System sends broadcast to User
2. System notifies User
3. User views broadcast

#### Use Case: Broadcast a message

1. Admin sends broadcasts a message to list of Users
2. System sends message to list of Users

#### Extension

##### 1a) Cancel broadcast

1. System discards broadcast

#### Use Case: Send Broadcast Message

1. Admin sends broadcasts a message to list of Users
2. System sends message to list of Users
3. System notifies Users

**End of Document**