Vent It

Scream away your frustrations.

http://ventit.xyz

Ang Zhi Yuan (A0150120H)

Marcus Koh Chi Hao (A0168472W)

Ngiam Ling Li Vivian (A0157464X)

Zhu Hanming (A0196737L)

Milestone 0

Singapore is a very stressful nation. From workplace-stress to stress from personal issues, Singaporeans are ranked as one of the most stressed people in the region. However, letting all this stress build up inside us is unhealthy. Prolonged stress may result in insomnia, fatigue, or in severe cases, depression and anxiety breakdowns.

We need a place to let all of this stress out, but how can we do so? We can't take it out on another person! We can, however, take it out on our phones with *Vent It*. *Vent It* allows you to de-stress by screaming your troubles away into your phone in a fun manner.

Milestone 1

Vent It's goal is very simple: to be an avenue for one to let their stress out. At the core of it, the web app allows users to type in whatever pain points they have, and let it out by screaming into the phone. The app will assign an "anger rating" based on how loud the scream was, and add it to a diary. Of course, ultimately our goal is to help people de-stress, so these pain points will eventually "fade away" and be archived as time passes (dependent on the anger rating).

It makes sense to deploy this as a mobile cloud application, because of the ease of access of such applications. Phone memory space is precious to users, and not a lot of people would be willing to download a full-blown app. Deploying it as a mobile application also helps our use case, as screaming into a phone is more natural than say, screaming into a laptop or a desktop to relieve stress. Phones are also highly portable, so users can go to a secluded corner to let it all out.

Milestone 2

I. Main Target Users

The main target users are students and young adults who have just entered the workforce, as these 2 groups are technology-savvy, and are subject to a lot of stress.

Students in Singapore are under immense stress. There is a constant pressure to do well from their family and teachers. They may also face relationship issues that might leave them frustrated without an avenue to let out.

Young working adults are no different. These people have entered the working world, and are facing things they have never met before in school such as office

politics. As the new birds in the office they will be under a lot of pressure to prove themselves in the working world. Similar to students, this group of people are highly likely to face relationship issues as well. *Vent It* will help them tide through these rough times by providing an outlet for them to vent their frustrations.

The Value *Vent It* Brings

Vent It leverages on how the act of screaming can be therapeutic and bring people a sense of relief when feeling frustrated. Some colleges even encourage students to let out a "primal scream" to help relieve exam stress. Screaming helps people feel better, even if it is only temporary. In addition, primal scream therapy is a legitimate form of psychotherapy used to treat anxiety, trauma, and stress.

By providing an outlet for people to release their screams and also possibly track them like a diary, *Vent It* encourages reflection, introspection and processing of frustrating emotions. Users will be able to look through past entries of "vents" should they want to and as time passes, these "vents" will be archived, encouraging users to let go and or process those emotions.

II. Marketing Plan

We will use a 3-prong marketing plan to onboard as many users as possible - Awareness, Retention, Reach.

1. Awareness

Since our main target audience are students and young adults entering the workforce, we would start promoting our app in schools. Exam periods such as the 'A' and 'O' level exams, university midterms and finals would be the focus timeframe to market and attract users.

Our **physical marketing plan** would include working with schools to place posters around the schools such as classrooms, canteen, toilets to encourage visibility. The poster will leverage on our cute angry mascot "Red" to first draw attention. With the rise of mobile-use and scanning of QR codes, there will be a QR code to allow immediate redirect to our website.

Sample of marketing poster

In addition, our **online marketing plan** would include four main channels, Email, Facebook, Instagram, and Twitter. To reach out to young working adults, we will work with university and polytechnic alumni offices to push out attractive Electronic Digital Mailer (EDMs) promoting *Vent It.* An email drip campaign will be employed to hopefully reach out to more people. Our social media strategy (Facebook, Instagram and Twitter) will leverage on our mascot "Red" to produce fun and interesting content such as a comic series, short animated stories on Facebook and Instagram and a personal rant page on Twitter for "Red".

On top of that, we will launch **Telegram and Whatsapp stickers** of our mascot "Red" to gain publicity of our app in a fun way. This will bring about awareness of our app and direct them to our site to try out.

2. Retention

To retain users on our app, we will employ an **email drip campaign** to prompt those users who have not used the app for a long time to come back and let out their frustrations. It will be tactfully planned out so as to not feel

like spam and drive the user away. The mascot “Red” would help in making the communications light-hearted and fun to attract users back on the app.

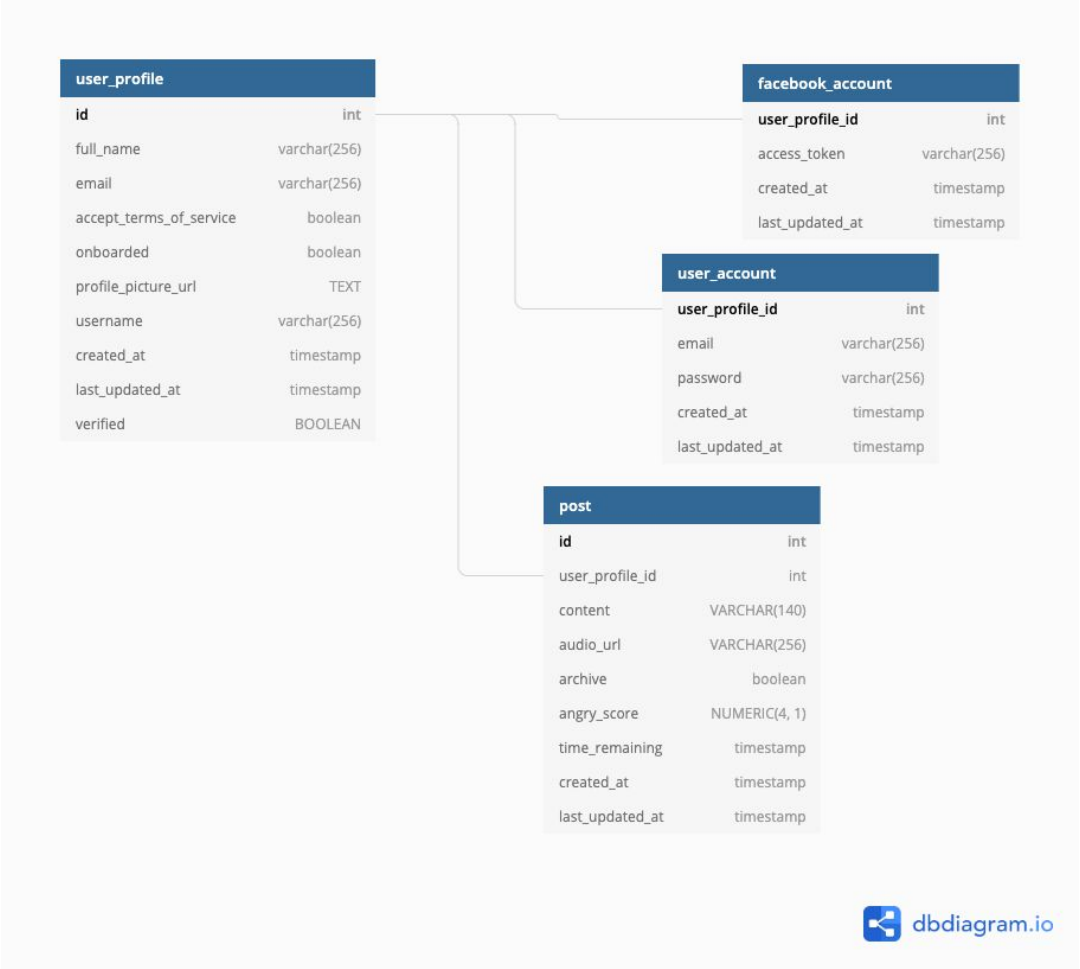
Moreover, we will carry out **giveaway campaigns** whereby users anonymously submit their most frustrating “vent” entry for a chance to win rewards like spa getaways, 1-for-1 drinks, 1-for-1 meals, etc. The “vent” entry with the most number of votes will win anonymously. These campaigns will be carried out on our site (for the submission of rants) and social media channels, mainly Facebook and Instagram. Such giveaway campaigns will keep the community engaged and encourage users to stay on the app.

3. Reach

Besides word of mouth, our **whatsapp and telegram stickers** of the mascot “Red” will encourage people to share with their friends. People might be curious and ask what is this sticker pack about and users can then share the app link for their friends to try out.

Our app is not a traditional product or service, making it harder to include incentives for a referral program. The nature of our app will encourage sharing if the user feels like the app has helped him or her release frustrations and has benefitted them emotionally and mentally. Thus, we will promote in this way - “If you found our app useful, share it with your friends!” or “Are your friends feeling a little down, frustrated and seems to be dealing with a lot? Share *Vent It* with them!”. By leveraging on the intrinsic value of our product, our **marketing strategy will take this direction** and hopefully encourage users to share with their friends who seem to be going through a tough time.

Milestone 3



Milestone 3.5

The primary key of the home_faculties table is matric_no, since a student can take double degree which means 1 student can belong to multiple faculties.

Milestone 4

REST API: https://www.getpostman.com/collections/9580e1d98e6eb7de2dab, https://ventit1.docs.apiary.io/

The API designed are easy to understand and readable. When an API is used to request something from the database, the HTTP method is usually GET. If the request is to

upload/submit something to the server, the method used is POST. Also, if the request is used to delete a post, the method used is DELETE. Furthermore, when getting a particular post, the route typically follows "/p/:id" where the "p" represents post and the id would be the post's id. Similar for uploading of post, the route would be "/p/upload" which means upload a post.

Milestone 5

INSERT INTO user_profile(full_name, email, username) VALUES ($1, $2, $3) RETURNING *

The purpose of this query is to insert a new user details into the "user_profile" table namely their full name, email and username and get back the data that was just inserted. This is used when a new user sign's up for Vent It.

DELETE FROM post WHERE id = $1 AND user_profile_id = $2 RETURNING *;

The purpose of this query is to delete a particular user's post using the post's id and the user's id and return the post that is deleted. The reason for returning the deleted post is so that the system can get the correct audio URL to be deleted from Digital Ocean Spaces bucket.

INSERT INTO post(user_profile_id, content, audio_url, angry_score, time_remaining) VALUES ($1,$2,$3,$4, to_timestamp($5 / 1000.0)) RETURNING *;

The purpose of this query is to insert a new post to the database with the user_profile_id, content, audio_url, angry_score and time remaining and it will return the post that was just inserted. The method "to_timestamp()" is to convert current milliseconds to Postgres SQL timestamp format.

The reason for most query with "RETURNING *" is so that we can have a single source of truth which is the database. Additionally, the $x, it is a placeholder for the values that are supposed to be in the query.

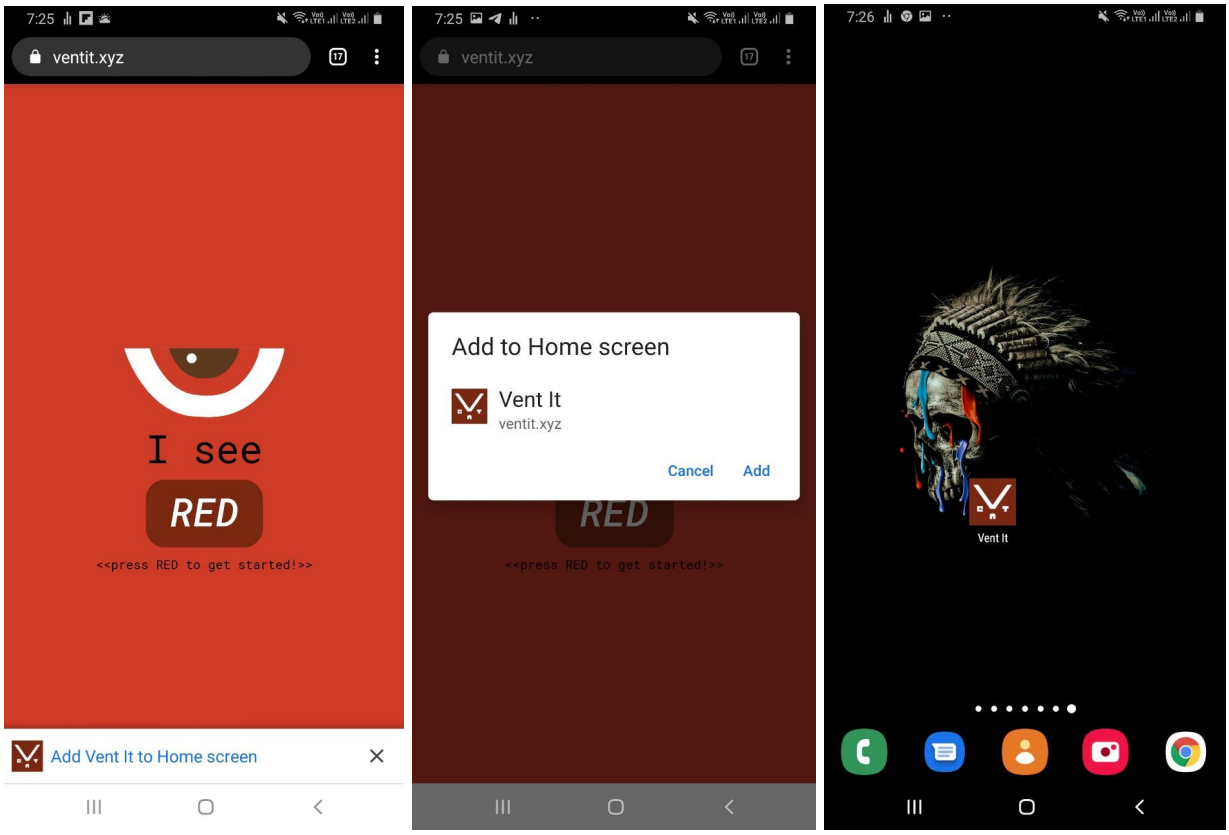
Milestone 6

Vent It's App Icon

Vent It's Splash Screen

Our icon design makes use of the word “vent” to form an “angry face”, making it memorable. To make our splash screen consistent with our icon, we decided to use the logo as the splash screen so as to condition users to our app and increase recall with consistency.

Adding application to home screen



Milestone 7

I. Reasons why our UI design is the best

Vent It's UI employs **"flat design 2.0"**. Flat design is a UI design style that uses simple, two-dimensional elements and bright colours. Often contrasted to skeuomorphic style that gives the illusion of three dimensions, flat design is a popular style that many big industry players in the market like Windows, Apple and Google are currently using. Flat design is great for responsive design and thus great for a PWA like this. By using simple shapes and minimal textures, flat design ensures responsiveness that work well and load fast, this is especially important for mobile devices. Flat design also provides users with a streamlined and more optimal user experience by keeping visual noise like textures and shadows to a minimal. Because flat design has certain drawbacks such as the absence of three-dimensional effects resulting in lack of cues for user interaction (e.g. buttons in flat design style may not appear distinct from other visual elements and does not appear clickable), we used

color variations - “flat design 2.0” - on major clickable buttons to increase depth and dimension, increasing visual variety and improving usability.

The original illustration of our **cute angry mascot “Red”** adds personality to the UI. Snippets of text from “Red” also brings in the element of humor as “Red” is a very sarcastic and “sian” character. These micro-interactions make the user interface so much more engaging and interesting to users. By using “Red” in our UI, it brings about a humanizing touch, elevating the experience from any other standard app.

To make it **simple and intuitive**, all clickable buttons are of consistent styling and colour so as to ensure visual consistency. When hovering over or clicking clickable buttons, the button will change colour so as to provide feedback to the user that the button has been clicked. This small colour change helps with the visual UI and thus usability.

II. CSS Methodology Justification

We used Bootstrap for our CSS sheets, and Bootstrap is very similar to multiple frameworks, employing similar conventions and standards to achieve stunning effects. As much as Bootstrap is not a methodology of its own, it is able to achieve much of what other frameworks seek to attain.

To a large extent, the various CSS frameworks out there boil down to making styling as modular and independent, so as to make it a lot more scalable. For example, OOCSS has its “separation of structure and skin” and “separation of container and content”, which Bootstrap also implements with its numerous classes, such as “text-white” and “btn-primary”, which aims to style specific elements in a single way. Through the application of these “single-functionality” classes, we can achieve the same effect as OOCSS using Bootstrap. This methodology allows us to be very specific and careful in our styling, applying the suitable styles at the right time. It also makes it very scalable, as we can style similar elements in various different ways, depending on the context and requirements.

At the same time, Bootstrap also allows you to easily understand the link between what you’re styling and the style you’re applying, with classes such as “text-red”. This helps greatly in understanding the classes, and though it’s nowhere nearly as semantic as it can be, it does make the CSS a lot more readable.

But most importantly, we chose Bootstrap for its highly modular utility classes, which greatly eases the styling process and helps reduce “aftereffects” from complexity in the code after repeated styling. Minimal custom classes are needed, as nearly all custom classes can be broken down into the various utility classes in one way or another.

Milestone 8

One of the best practices for adopting HTTPS is to use a strong certificate with a large key-size (e.g. 2048-bit), as small key-sizes are prone to brute force attacks. Then, we need to implement a server side HTTP redirect that redirects all non-secure requests to a HTTPS version, as well as a HSTS header to force HTTPS connections for the hostname. Finally, a chain is only as strong as the weakest link, so we also need to make sure all our dependencies are HTTPS-enabled as well.

Certificate pinning works by associating a given host to a certificate or a public key. Then, on subsequent visits by the client to the website, the client will verify the website's certificate against the pinned certificate or public key. This will help protect the website from malicious attackers who attempt to spoof the website with the same hostname.

The downside to certificate pinning is that if the certificates change, the app must also be updated to reflect the new certificate. If public key pinning is used, the protection will be nullified if the upstream certificate signer gets compromised (which has happened before with DigiCert).

We opted not to use certificate pinning as it would require us to update the app everytime we renew the certificate, and the Let's Encrypt certificate has a very short validity period of 3 months.

Milestone 9

Currently, the only offline functionality is the caching of the user's past "vent" entries on the Home screen, individual "vent" entry screen and the archive screen. This allows users to revisit the vent entries that they have made previously and reflect on the times that they got upset, both in terms of chronological order and in order of how angry they were. This fits the user expectation is because even without an internet connection, when the user opens the app, he/she would expect to see some content on the home page and not an entirely blank page. This is because in apps like Facebook or Instagram they cache their images and text aggressively and even without an internet connection, user is still able to browse the app. We have decided to mimic this feature and allow our user to browse the vents that they have posted.

The cache / localStorage is kept up to date as frequently as possible, and an update happens every time the user loads either the Home screen or the Archive screen with online access. After which, when the user goes offline, the "vent" entries are kept the same way until the next time they go online.

The main case that we wanted to take into consideration is that any addition to the user's "vent" entries through alternative platforms would be synchronised and fetched as soon as possible.

The user will not be able to make modifications to the content once they go offline, and that includes any addition and deletion. The reason for this is because currently it would waste a lot of storage space of the user to cache a 5 second audio file. We would need to convert the audio file into a base64 array and store that array as a string within localStorage. This would drastically increase the total amount of storage the user would need to use for our app. This is not ideal and which is why we have decided to not support it within our application. However, as PWA progresses and allow for more file types to be stored, caching audio files would definitely be a go.

Milestone 10

Session-based authentication works by storing the session state on the server. Upon login, the server would create a new session state and pass a random string to the client which would be used in every request made to the server. The server would verify this string against its own copy and process its internal session state accordingly.

Token-based authentication, on the other hand, is stateless. Upon login, the server sends a token to the client. This token contains the user's information and capabilities, and is signed by the server. Unlike session-based authentication, the server would verify this token by verifying its signature.

As session-based authentication requires a session state, it requires more resources on the server end to process and maintain every session. Token-based authentication, however, trusts the token completely, so there is no requirement for a state, and therefore less resources for the server.

Session-based authentication also has the problem of server switching. For larger applications that are serviced by multiple servers, the developers must find a way to maintain the session state across all servers in case the user were to switch servers for load balancing purposes.

Our PWA operates using a REST API, which in itself is stateless, so it doesn't make sense to keep a session on the server and waste the server's resources.

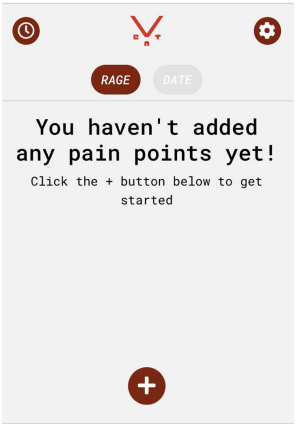
Milestone 11

The reason why we chose Bootstrap is because as an anger app, we wanted the UI to be flat and simple. Additionally, most of the developers in the team were familiar with

Bootstrap as well. The reason why we did not choose Material UI is because it did not provide flat UI styles rather most of their design uses shadow as an indicated of how far each UI object is from one another.

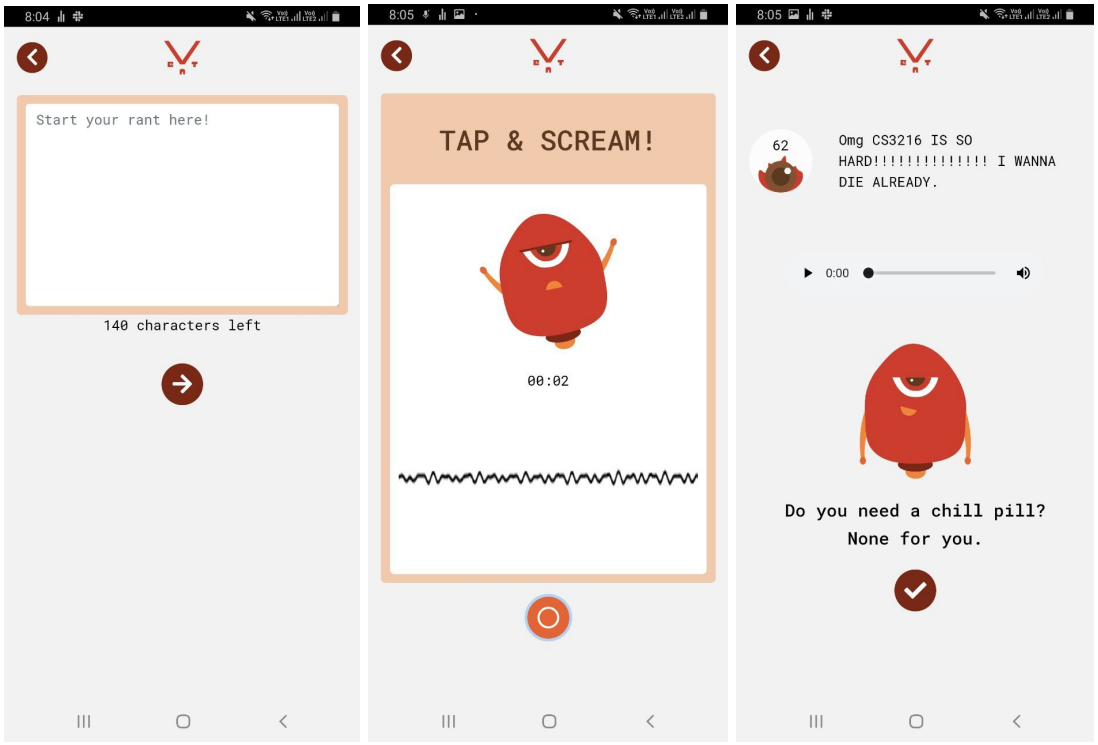
I. Calls-to-action front and centre

The main part of our application is placed in the most prominent part of the screen. Previous posts are listed in the centre, and the button to add posts is centered at the bottom.



II. Streamline Information Entry (upload form)

Within the app where you can upload a new Vent, we segment the process of uploading a new Vent into a 3 step process. This makes it easy for the user to follow through the steps and easy for the user to understand what to do at each page. This way the user would not feel frustrated than they already are as the 3 step process is easy and simple to understand.

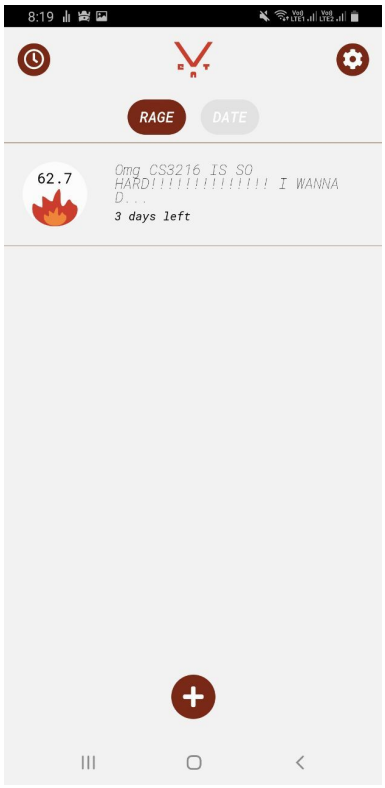


III. Minimise form errors with labeling and real-time validation (text entry)

When a user is creating a new Vent, the user needs to input some text before they could proceed on to the next step. If they tried to proceed to the next step, they are shown an error “You need to enter something!”. This provides real time feedback to the user allowing them to understand that they cannot proceed to the next step if they do not type something.

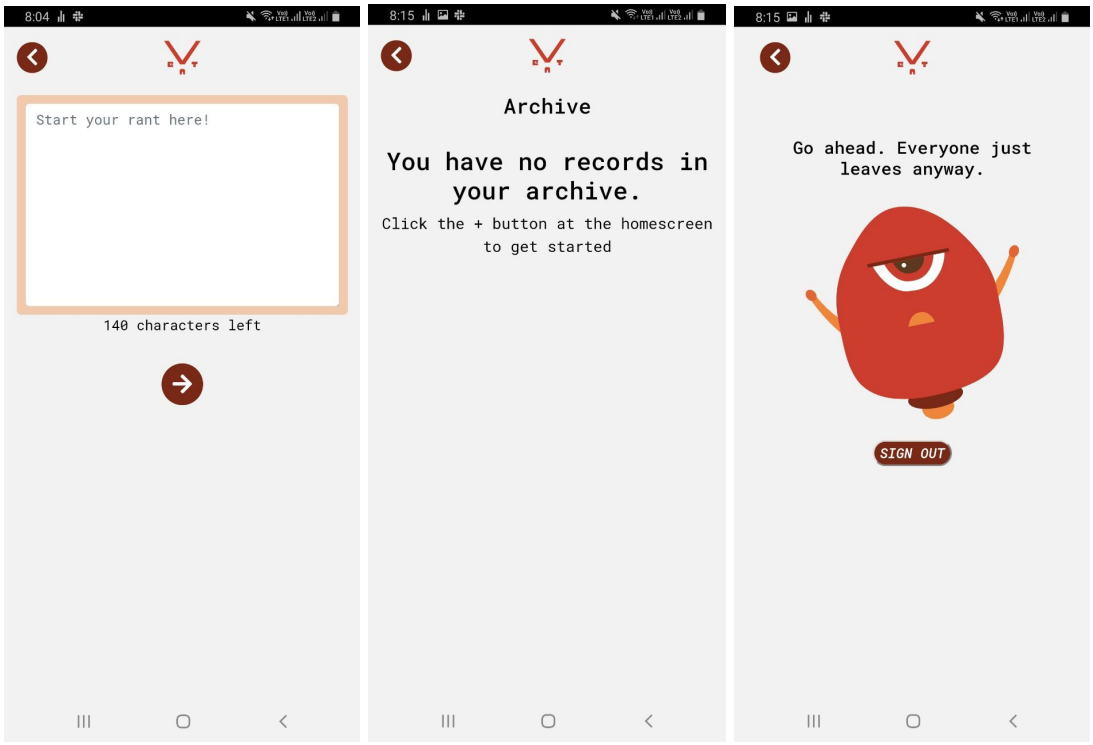
IV. Keep your user in a single browser window

Our PWA is designed in such a way that every interaction with the website is stayed within the website. The reason for this is because we followed the concept of Single Page App (SPA). We do not link the user to other browsers except for when they are attempting to login via Facebook or confirming their emails. Besides those 2 workflows, the user stays within the application and is not routed anywhere else unnecessarily.



V. Make it easy to get back to the home page

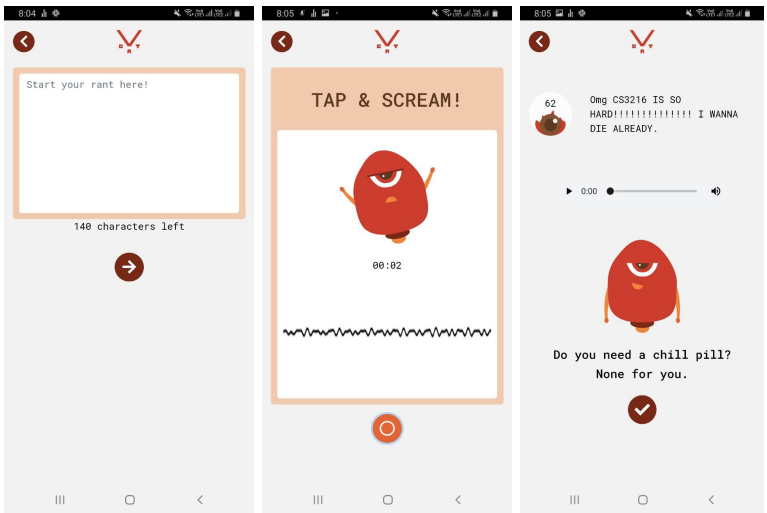
Within the application, we provide easy to access back buttons at each screen where the user can easily click on the back button and simply go back to the home page. This experience or behavior is widely known to many users on both iOS and Android. By providing this back button, the users would immediately know that it would bring them back to the previous screen which is the home screen in our case.



Milestone 12

I. Adding an entry

Adding an entry is broken down into 3 simple and easy steps rather than having all done in one single screen - typing your rant, recording your scream, and saving your “vent” entry. This reduces clutter and information overload, allowing users to take very little effort to complete these steps.

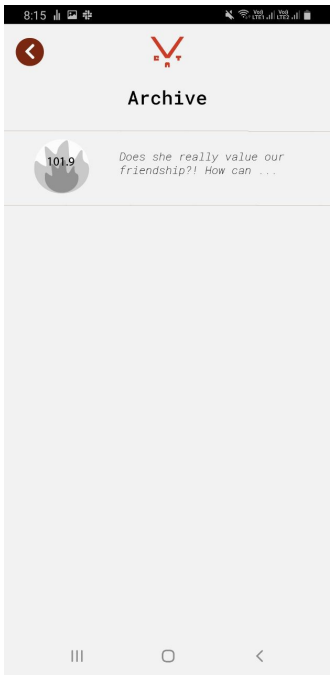


II. Scream recording

We capped the scream recording to 5 seconds as it is very tiring for a person to scream more than 5 seconds and we want to keep it short so that users can let go of their frustrations in a short and good scream. Also, we don't want users to be talking into the phone like a voice recording function where there is no time limit as it defeats the purpose of a scream outlet. To encourage short bursts of venting frustrations, the 5 seconds cap is enforced in our user interaction design.

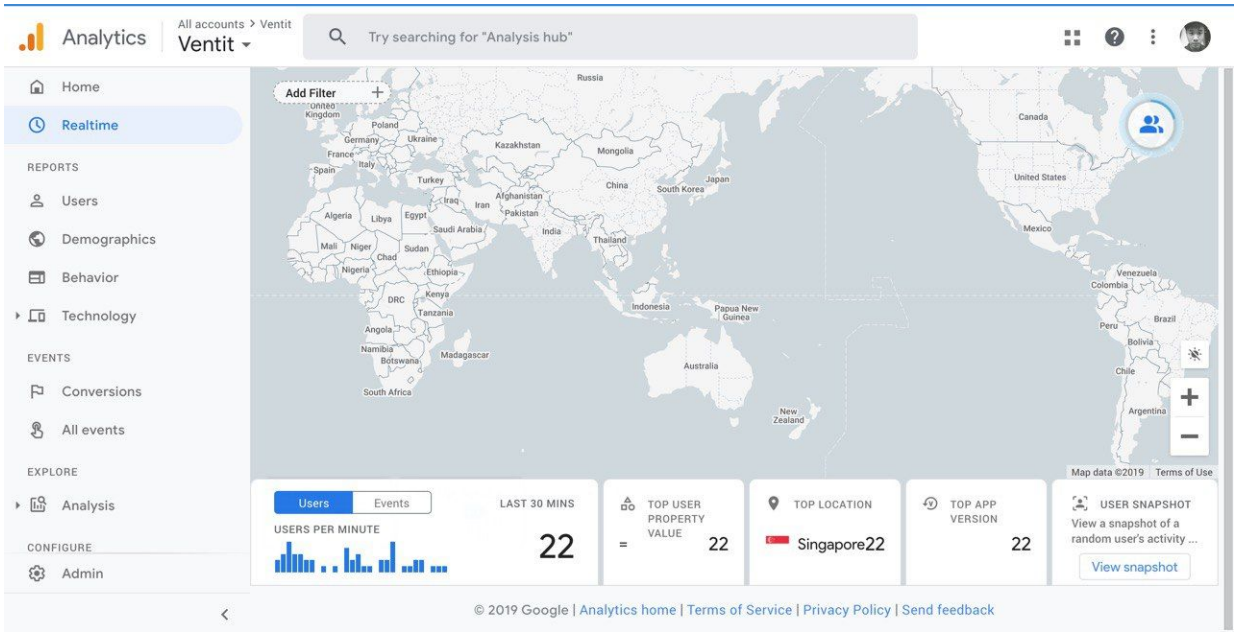
III. Archive

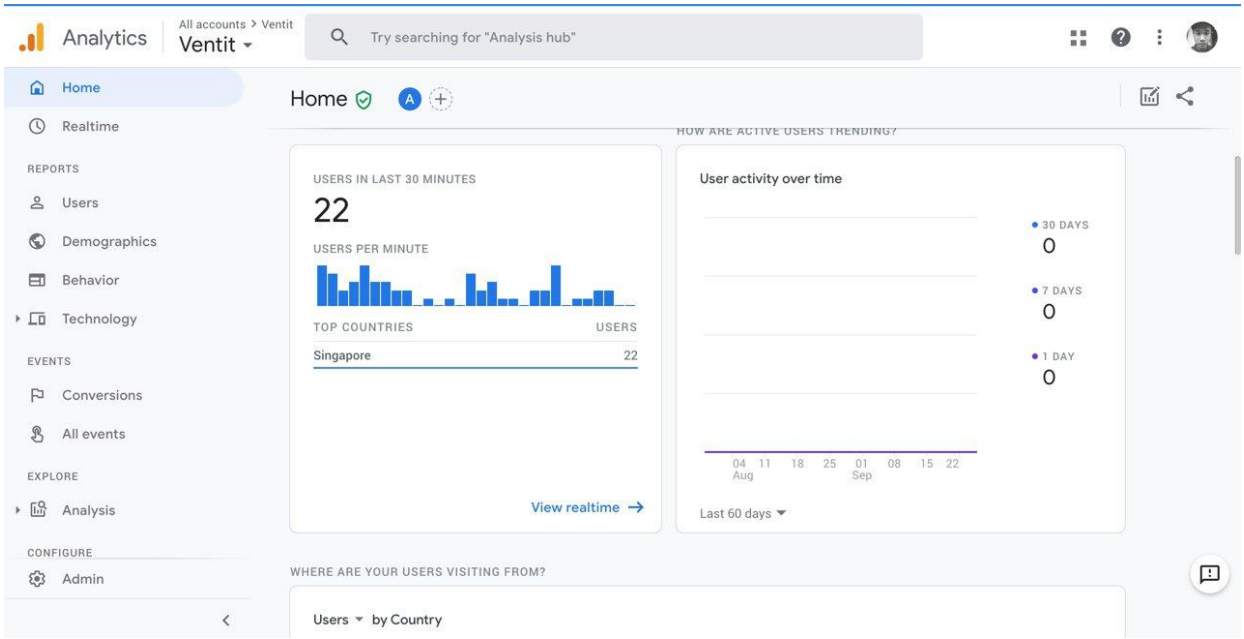
"Vent" entries go into archives automatically based on the anger meter associated with the entry. For example, the higher the anger meter of the "vent" entry, the longer it will stay on the homepage. When they go into their archives, they are able to view past entries, however the entries are "greyed" out because they have faded. Making this function automatic rather than manual was to encourage "letting go" of these anger as time passes. Sure, the user could have more control and manually archive each entry when they are ready to. However, rather than focusing on the action of archiving, we wanted users to have one simple action which was to just vent. We wanted a focus on being an outlet for them to release their frustrations. The automatic archiving also brings about a different user experience since this function is more of a secondary function whereby the user can view past frustrations should they wish to without even needing to make an extra step to archive entries. The time left before the "vent" entry goes into archive is also shown on each entry to provide users feedback on the automatic archive concept.

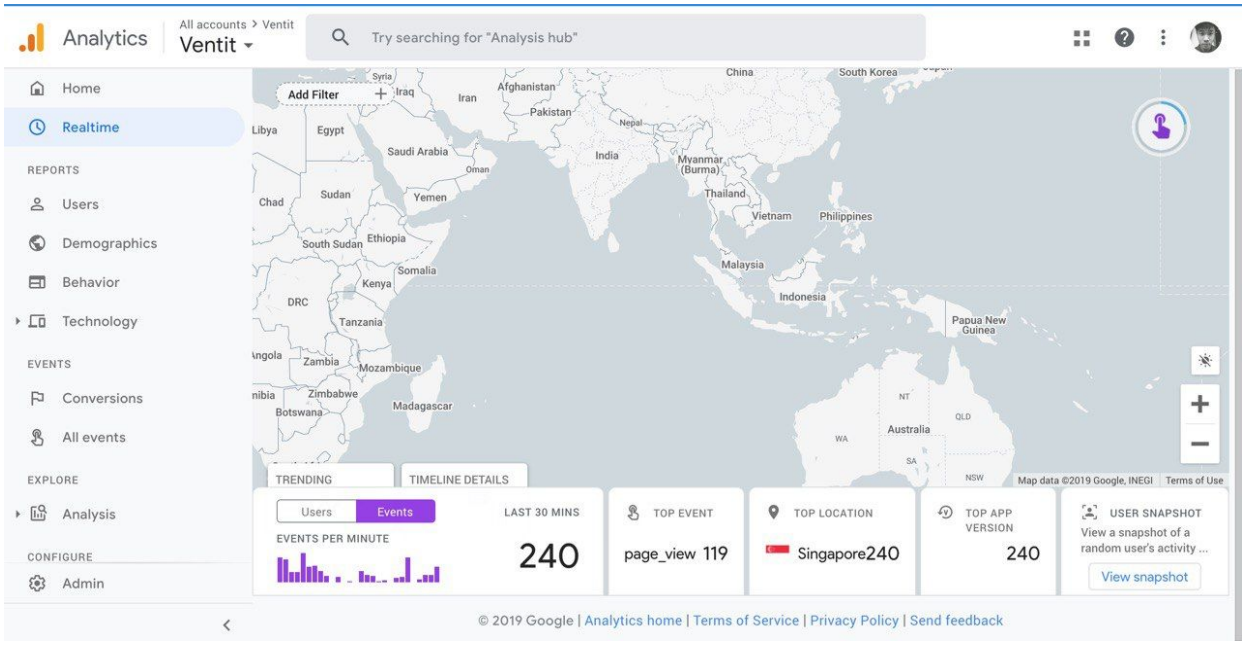


Milestone 13

Google Analytics screenshots







Milestone 15

Our app allows users to use their Facebook login to authenticate to our service. This lowers the barrier to entry, as users can login with a single tap as opposed to having to create an account. The reason why Facebook authentication was chosen over other login platforms such as Google login or Twitter login is that the majority of the main target audience are teenagers and young adults. Most teenagers and young adults in Singapore have social accounts such as Facebook that they use frequently and it would make sense for them to login with something that they are familiar with and they would definitely have an account with.