ASSESSMENT 3 - PROJECT

Sharper Tech

Sharper Tech
Evan Phelps (s3934101), Gavin Jamieson (s3925654), Rory Henderson (s3933584), Kyle Ross (s3930665).

The information in this report is also duplicated on our group website which can be viewed HERE [https://jamgav.github.io/Assessment3/index.html

You can also view our Git Repository HERE [https://github.com/JamGav/Assessment3]

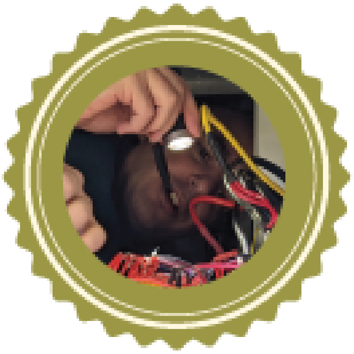
Contents

Team Profile

Sharper Tech is comprised of Evan Phelps (S3934101), Gavin Jamieson (S3925654), Rory Henderson (S3933584) and Kyle Ross (S3930665). Sharper Tech, our team's name, is an anagram created by using the letters of our members' names to create “Sharper”.

Personal information

GAVIN JAMIESON
S3925654



My name is Gavin, I am an Australian who grew up in Brisbane and lives in Sydney. Currently I am studying Introduction to IT online via OUA and working on team Sharper Tech. I have a keen interest in the world around me and learning about why things are the way they are. My interest in IT stems from a hobby interest in electronics and programming which initially drove me to pursue a trade in Aircraft Electrical Systems (Avionics).

Throughout the years I have tinkered on and off with a number of different electronic and IT-related projects while finding a keen interest in projects that bridge the gap between the physical and the connected world, and automation projects. As someone who is not a huge fan, contributor or user of social media, one of my favourite builds was developing my own blog for my travelling based on Wordpress. I was able to develop a unique page on it that would receive position updates via my HF radio and post them on a map along with short tweets so that family and friends with the link could follow along.

RORY HENDERSON
S3933584

I’m Rory and I am the R in Sharper Tech (one of them...). I was born in Scotland, grew up in Brisbane and now live in Western Australia. In 2019 I was thrown into IT and now coordinate IT for 2 campuses with around 350 users. I have no formal training in IT and am pursuing a Bachelor of IT at RMIT as it offers a great base, giving an understanding of core IT concepts with the ability to specialise in specific areas later in the degree structure. The flexibility of online study and lighter load than a traditional semester structure is appealing whilst I continue to work.

Although I have no formal training, IT has been a part of my life for a long time. My father is a web developer and always encouraged us to explore IT. Computer games were a big part of this interest when I was younger, although I was also interested in Audio/Visual productions which allowed me experiences such as building PCs capable of 4k editing, running multi-terabyte shared network drives and managing multi-camera live streams. I also work as a bar manager and create video graphics for theatre shows. My recent projects include The Boy from Oz and Priscilla, Queen of the Desert.

Evan Phelps
S3934101

G'day, my name's Evan, and I hail from all over the place. Born in the UK, but grew up living several years at a time in various Australian towns and cities, especially Perth and Alice Springs. Alice was where I first picked up a guitar during early high school, and since then that hobby has grown into a passion, and from there it lead me into a full-fledged Music Performance degree in the UK. After that was done, I came back to Aus and settled down in Tasmania, ready to pursue new things.

I've always had a very definite interest in the field of IT and programming, what with computers and software being the place I've spent most of my time. Hoping now that this course can gradually take me down the path of concrete career skills, and secure a solid future, and to that end I've joined up with Sharper Tech, proud to be a contributing team member.

Kyle Ross
S3930665

Hi, I'm Kyle, I'm the other R in Sharper Tech, I was born and have lived on the Gold Coast my whole life. I am currently studying Information Technology through Open Universities as a way to test the waters of further IT study while I continue to work from home part-time. Nearly all my hobbies involve computers, with them being video games, music (where my taste is all over the place), movies, or just messing about on the internet. These hobbies spawned my interest in IT and kept it alive while my high school did its best attempt to crush the interest.

I'm most interested in the area of programming, where I'd love to learn to make my own applications or games. The ability to work for yourself if the opportunity presents it is a very tempting one and one of my favourite perks of the Information Technology industry. Outside of my usual computer-centric hobbies, I enjoy spending time with my mates doing whatever catches our attention, watching various Philadelphia based sports teams (go Sixers!), reading random literature that catches my eye, or at the gym where I train 5-6 nights each week.

Group Processes

The group worked cohesively for Assessment 2 and brought the same can-do attitudes and work ethic into Assessment 3. One change that was made was a stronger review process of work that wasn't our own. What this meant was empowering each individual team member to make changes as they saw fit to work that someone else had committed. Due to the work being committed regularly to the repository, any changes were able to be seen in the blame log and either challenged or supported as required.

Another change was our approach to team meetings. During A2 we elected to share the Role of Chair and scheduling around the group, which allowed different members to experience the role, but also lead to uncertainty and inconsistent structures for Agendas and Actions. For A3 Rory scheduled a weekly recurring meeting for Wednesdays after our course tutorial. He chaired the meetings and also managed the agendas, meeting minutes, action items and any additional meetings that needed to be scheduled.

With one group member sadly having to move on to other things, the group was able step it up another level and divide the workload amongst the remaining members based on their own individual strengths displayed from Assessment 2. The result was work delivered to the same standard within the time constraints of the project and life commitments.

Career Plans

Games Developers - Kyle

Kyle has a desire to pursue a passion in game development. His desire to work in Game Development requires a thorough knowledge and proficiency in coding languages such as C# or C++, as well as the knowledge and understanding of game engines such as Unity or the Unreal Engine. It is also expected to have experience in general software development as the work is heavily code related.

Kyles career plan and ideal job has remained unchanged throughout this project. His passion for computing lies within the creation and contribution to video game development. During this course he was able to uncover what skills are required to fulfil and acquire that role, and thus plan to further his understanding of game engines, as well as my proficiency in programming.

Back End Developers - Gavin and Evan

Evan and Gavin both listed roles that fit under the description of backend developers. Their desire to work in backend development requires skills across a multitude of disciplines in the IT sphere, including programming in a variety of languages, database development, cloud management, and networking.

After some self reflection, Gavin has indicated that he will pursue a role that combines the technical side of IT with the business analytics side. This may lead on to a Data Science-type role as there is significant crossover of the fields. Further study will be required.

Thanks to the research done throughout this project, as well as what was studied throughout the general coursework, Evan has opened his mind to the wider possibilities of backend development, as well as backend-adjacent areas like databasing. The prospect of carefully working to build these interlocking systems has proved quite enticing.

Azure Incident Manager - Rory

Rory's ideal job was as an Azure Incident Manager. His desire to work as an Azure Incident Manager requires extensive cloud computing knowledge with a focus on the Azure product line, as well as the ability to communicate well with his team, to be able to work under pressure, and the ability to debug and remediate issues.

Over the study period, Rory's knowledge of cloud platforms and understanding of their growing importance in the future of IT has reinforced his interest in this career path. Before the course he only had experience with the Microsoft cloud platform offerings but is now also familiar (in theory) with the Amazon and Google offerings, broadening his interest away from just Azure.

A3 and A5 also gave Rory the ability to function in a team management role, serving as the project lead and chair and the tasks.

Career Similarities

Our desired jobs often overlapped in what skills we are expected to possess, such as Evan and Gavin's ideal jobs of backend developers. As game developers are also classified as backend, they all required proficiency in skills such programming, databases, and networking. Though Rory's ideal job was an outlier to the rest of the group, his desire to work with Azure required cloud computing knowledge, something that is expected from Gavin and Evan with their desires to be backend developers.

Overview

Topic

Pursuit of Hoppiness is a community-sourced bar rating and review app that can be used to filter by specific categories to allow beer enthusiasts to broaden their beer horizons, beer bargain hunters to find and deal, and all beer lovers to come together in one place. It is the “Pursuit of Hoppiness”.

The app focus on reviews of venues and encourages exploration of a user's local area or can serve as a guide when you arrive somewhere new, providing 5 different beer-focused categories to filter by instead of a single vague 5-star review.

Motivation

Beer is the most consumed alcoholic beverage in the world and third most popular beverage after water and tea (Salanță et al., 2020). Despite this, when one explores new bars, they frequently find the same 4 lager taps installed. “Pursuit of Hoppiness” solves this by quickly sorting all nearby venues based on Tap Uniqueness, Total Taps, Pint Price, and Atmosphere based on fellow beer enthusiasts' input. The app also allows reviews and social networking with friends.

The choice to develop a Mobile Application was made to reflect the continued growth of the mobile sector. In 2022 there are 6.5 billion smart phone subscriptions with this expected to grow to 7.7 billion by 2027 (O'Dea, 2022) and 88% of the usage time on the devices being within apps (Wurmser, 2020). Members of the craft beer community are also already using mobile applications to enhance their experience with 38% reporting app usage (Kelsey, 2016) so the choice to develop for mobile was obvious.

Any potential project experience would be relevant to future employers and although this project may not be the worthiest, its development touches on several skills and specialist areas. It also has proven to be a good conversation starter and would add a memorable human element to any job application.

Landscape

38% of craft beer drinkers in Australia use beer apps (Kelsey, 2016), equating to a large market and many competitors. There are many similar applications on the market, but the use of our 5 categories of venue filtering, 6 including distance and social elements, brings something new not offered by others in the space.

Untappd is the largest existing competitor in the market. They are a craft beer review app on iOS and Android with a social element. The app allows users to check into venues, try and review beers, tagging them to that specific location (Untappd, n.d.). The app had 8 million users, 900 million venue check-ins and 20,000 businesses as of mid-2020 (Smith, 2021). Similarities between our concept and their product include the map UI, social elements, and badges. However, our concept differs with their product exclusively focused on reviews of beers, whereas ours will focus on reviews of venues.

Untapped

Beermenu

Untappd's business model charges bars and breweries an annual fee for membership, providing them features including digital and print menu web-based design applications and hosting, access to specific customer groups, event marketing, push/email marketing, social media integrations, and access to a beer database. There are other apps that use the same model, including Beermenus (Untappd for Business price increase, 2022). Giving bar owners control of their location is a feature we are planning to include, but we had not considered monetizing this feature.

The app MisterGoodBeer is another product on the market with similarities to the Pursuit of Hoppiness. It shows nearby locations and can be filtered by price, and happy hour. Contributions to the app are made by the users. The app has over 151,000 downloads and 12,400 contributors (MisterGoodBeer - Les bières pas chères autour de soi avec l'happy hour, n.d.). The app features a similar UI and venue focus but lacks the social element that we plan to bring.

References:

Beermenus.com. 2022. Untappd for Business price increase. [online] Available at: https://www.beermenus.com/blog/312-untappd-for-business-price-increase [Accessed 20 January 2022].

Kelsey, R., 2016. 2016 Australian Craft Beer Survey Results. [online] Beer Cartel. Available at: https://www.beercartel.com.au/blog/2016-australian-craft-beer-survey-results/ [Accessed 8 February 2022].

MisterGoodBeer. n.d. MisterGoodBeer - Les bières pas chères autour de soi avec l'happy hour. [online] Available at: https://www.mistergoodbeer.com/ [Accessed 22 January 2022].

O'Dea, S., 2022. Smartphone users 2026 | Statista. [online] Statista. Available at: https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/ [Accessed 17 February 2022].

Salanță, L., Coldea, T., Ignat, M., Pop, C., Tofană, M., Mudura, E., Borșa, A., Pasqualone, A. and Zhao, H., 2020. Non-Alcoholic and Craft Beer Production and Challenges. Processes, [online] 8(11), p.1382. Available at: https://www.mdpi.com/2227-9717/8/11/1382/htm [Accessed 2 January 2022].

Smith, C., 2021. Interesting Untappd Statistics and Facts (2019) | By the Numbers. [online] DMR. Available at: https://expandedramblings.com/index.php/untappd-statistics-facts/ [Accessed 21 January 2022].

Untappd. n.d. Untappd. [online] Available at: https://untappd.com/home [Accessed 22 January 2022].

Wurmser, Y., 2020. Apps Far Outpace Browsers in US Adults' Mobile Time Spent. [online] Insider Intelligence. Available at: https://www.emarketer.com/content/the-majority-of-americans-mobile-time-spent-takes-place-in-apps [Accessed 16 February 2022].

Aims

The primary aim of the Pursuit of Hoppiness is to improve our users' beer drinking experiences and encourage exploration. This will be achieved by creating a beer-focused, location-based, bar filtering system that can be tailored for what the user is looking for at that time. The idea was initially inspired by a poor experience at a bar that had high ratings on Google Maps but wasn't what we were looking for that night. The Pursuit of Hoppiness would have made that night better, allowing us to not just see a single rating out of 5 stars, but specifics on our 4 categories, Tap Uniqueness, Total Taps, Pint Price, and Atmosphere.

To achieve this core functionality, we need the ability to record users reviews of our categories with the ability to quickly serve this information as an average to users and allow them to filter venues based on it. This information would also ideally be tied to a map UI to allow easy navigation. Map/location data would also be used to provide users with relevant venues (distance wise) and the ability to search other locations of interest. Additionally, location data would be used to validate the authenticity of submitted reviews ensuring the reviewer is proximal to the venue and existing map data would be pulled to populate venues which haven't received user reviews.

A strong community is an important pillar to the application. It is crucial as we require user generated content to achieve functionality. Our initial prototype heavily features user reviews and individuals' profiles by displaying their reviews prominently on the venues page with a significant portion of this space given to the user's profile picture. This page also encourages users to contribute with a “Rate and Review” button. Users' personal profiles also place emphasis on their contributions with a timeline of their reviews.

We plan to integrate all of these features into the initial release of the Pursuit of Hoppiness, but have plans to develop it past this core functionality in the future.

Developing the community experience of the Pursuit of Hoppiness is a major goal after completion of bar filtering on our categories. Introduction of social elements, such as friends, a friend activity timeline, chat and sharing would be a major focus after the initial launch. As community contributions are essential to the app, developing features that support the community are a major focus.

To encourage engagement from users, introduction of badges would also be quickly implemented. Badges would be awarded for behaviours that benefit the app. For example, for users who contribute regularly or reach contribution milestones, make connections with other friends or regularly use the application. Badges would be made visible on the user's profile picture to encourage competition between users.

Introducing venue manager controls is also an important development goal, both to improve functionality for our users, but also to allow monetisation. By providing venue managers with the ability to respond to user comments and a dedicated venue timeline they can engage with users and help promote their venue. Features such as venue view statistics and the ability to promote their venue also introduce a monetisation model which we found is being applied by Untapped and Beermenus which are two similar applications already existing in the market.

Plans & Progress

Initial Data Seeding

As the project's Phase 1 aim is limited to the Perth area, we must be able to provide data to the users to kickstart organic growth. Initially it was thought that web information could be scraped from a source like Google Maps, but this prompted some ethical considerations once it was realised that the same thing could happen with the data that we are holding. Consideration is given to budget when deciding how best to collect this data. Pursuit of Hoppiness is a small operation and can ill afford to send its staff out on beer hunting missions during the limited available time to produce a proof of concept product. A number of market research companies exist within the Perth area that support the hospitality sector. This work could be outsourced to a market researcher and then ingested into the database to provide a starting point for users.

Google Maps Technology

Selection

Google Maps is the preferred map application to use for this product. Google Maps is chosen as it is the most widely-supported across multiple platforms and most users will be familiar with its look and feel. It is important that we create a common feel for Pursuit of Hoppiness across all platforms.

Other considered options

Alternative map products such as Apple Maps and Bing maps were considered however they were discounted fairly early on. This was largely due to the requirement to reduce the need for different streams of development. As Google maps can be deployed with a similar look and feel across a variety of platforms, selecting Google Maps allows us to focus development in one area. As google maps is the most popular navigation app(Ceci, 2022), developers in the mobile space are also more likely to be familiar with the API.

Prototyping

In the initial prototype phase of the product we aim to deliver a small mockup based on static data showing the locations of a number of venues with a brief description. This will help us to decide what important bits of data we can safely display on a mobile device without the screen becoming cluttered.

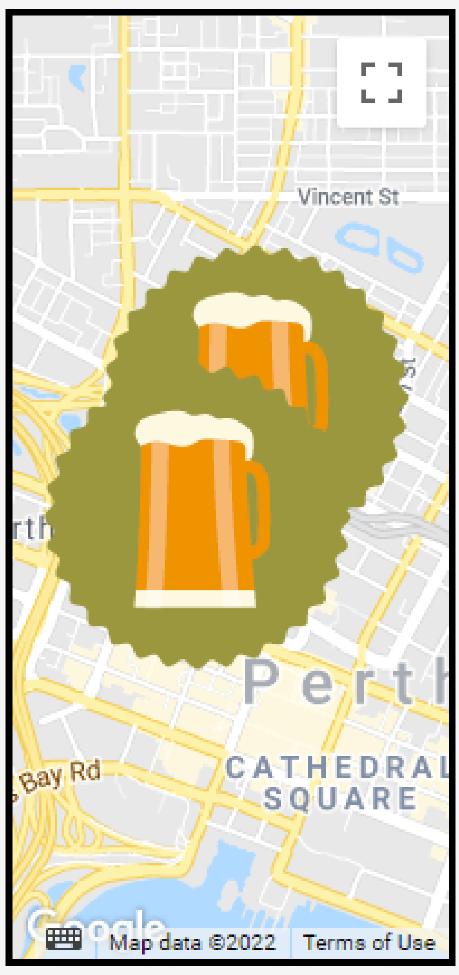
For the purposes of the demo, the API key for Google Maps has been committed to the public repository. The mitigation for this risk is to heavily restrict the key in the Google Cloud Console to the HTTP referrer of the site. While this means that testing may be impacted between individual users, if they need to test on a local machine, they can replace the existing key with a private key of their own and enact their own controls to prevent pushing it to the repository.

The mockup data is currently static. For the closed alpha release the code will have to be developed to allow incorporation of data from the REST API.

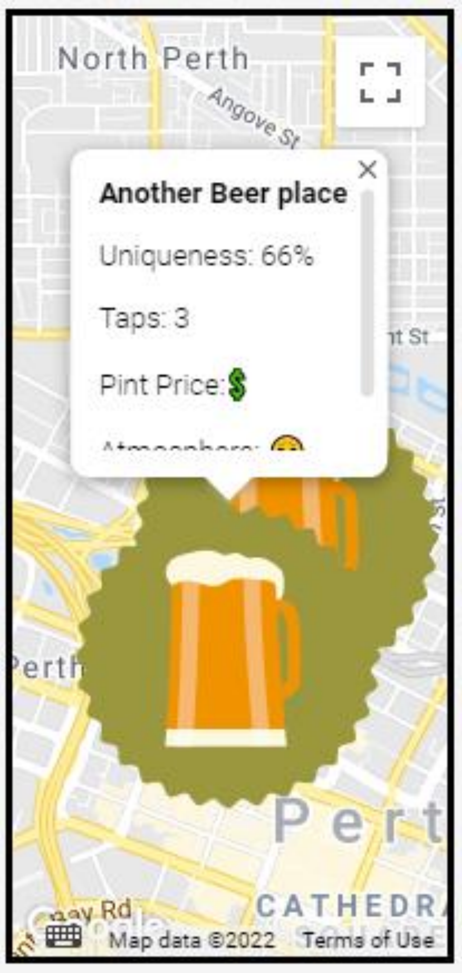
The anticipated parameters in the request to the database for map data are Map Zoom Level and Map Centrepoint. This information will drive a query that will feed data back to the client using our API in a format like the following:

```
{
  0: {
    "venue": "Another Beer place",
    "position": [-25.000, 153.000],
    "unique_score": 5,
    "tap_count": 4,
    "pricing": 6,
    "atmosphere": 8
    }
}
```

Storing the unique score, pricing and atmospheres as integers will allow flexibilty going forward in how the information is presented to the user. For example, "atmosphere" with a value of 6 may be represented along a slider or by a colour scale depending on the direction chosen for the app based on feedback.



Navigation View


Another Beer place

Uniqueness: 66%

Taps: 3

Pint Price: $

Atmosphere: 😊

Venue View

Please visit https://jamgav.github.io/Assessment3/project.html for an active Prototype

Post Prototype

Post-prototype and for Phase 1 we aim to integrate this with data supplied from our backend database. This information will be fetched by the relevant frontend application using our API and then displayed on the device. As Google Maps supports a JSON input, this is what our API will provide. It is intended that the front end will provide the location of the map view; this will be passed to the database query which will return a dataset to be formatted into JSON with relevant information for display on the map.

Significant refinement of the mockup will be required to make it match up with the layout envisioned in the UI development section

Hosting

Early phases

For initial testing to reduce costs we have opted for a local machine to host the application data and to deliver the content, the "box under the desk" approach. A Virtual Private Server arrangement was investigated, but after comparing costs it was decided that a migration to an off-premises solution would only be appropriate after the completion of the closed alpha testing phase.

Once the decision is made to move to the open beta stage of development, the application data, content, and code will be moved to a cloud hosting provider such as Google. As our development will be completely containerised, it will be a trivial matter to migrate the entire project to a cloud hosting provider that best suits our needs.

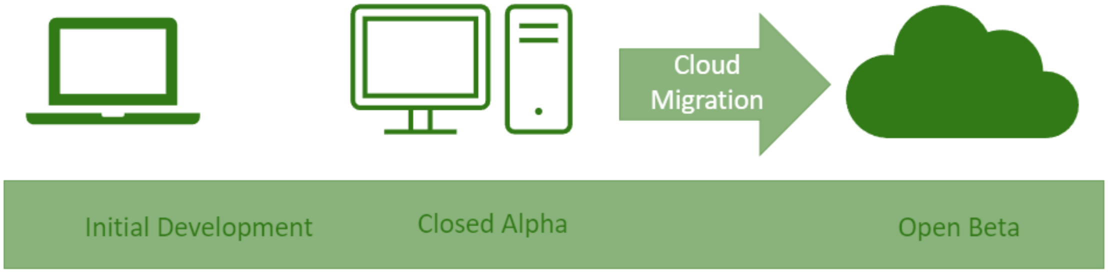
Selection of a Cloud Service Provider

A variety of providers were investigated as were a number of ways of hosting Pursuit of Hoppiness. Between Platform as a Service and Infrastructure as a Service architecture it makes sense for us to adopt an IaaS setup as this will reduce our need to manage the operating system and instead focus on upkeep of the application and software supporting the application.

Google, Amazon, and Microsoft are the big names in this space all offering a product that meets our needs. They all also provide data centres located in Australia providing low latency times nationwide. Due to us already adopting Google maps as a service within Pursuit of Hoppiness, Google is a likely frontrunner for selection as it will allow us to keep all billing in one place and reduce complexity. While we must remain focussed on cost control it may be worth paying a small premium to reduce complexity and risk rather then increasing both to save a small amount.

Cloud migration

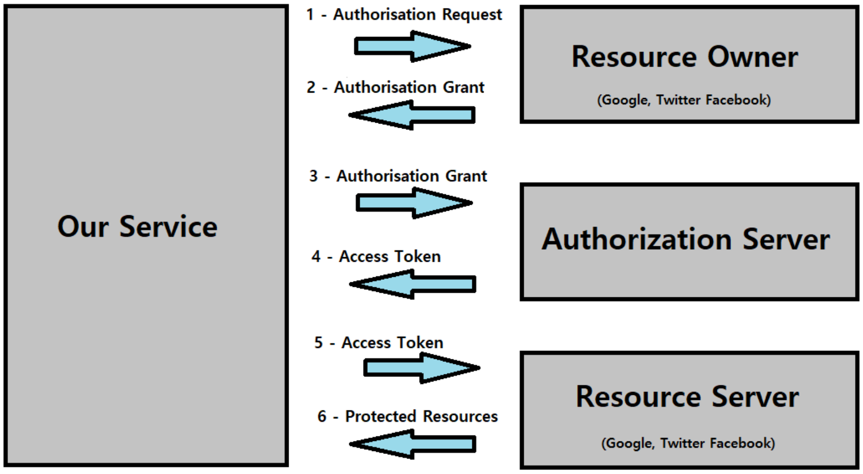
Our cloud migration signals a significant milestone for Pursuit of Hoppiness. This allows us to extend our reach and further reduce the risk of an outage affecting our users. Cloud migration is expected to be a quick process with the entire application and database being migrated over the course of a single day with minimum outage.



In summary, the application will first be developed on local machines, then hosted in-house for rapid testing of deployment pipelines and to reduce costs, then moved to a cloud hosted solution. This will all be based upon certain milestones being reached within the project.

OpenAuthorization & Logins

When creating an account with our service, we decided we would implement OpenAuthorization so as allow users to join the site through existing accounts such as Google, Facebook and Twitter. Users are also offered the choice to join our application via the standard procedure of creating an account through email and password.



Account Data Storage

OpenAuthorization Storage

When creating an account with our service using a third-party application and OpenAuthorization protocol, the third-party service that they sign up with (e.g. Google) will supply OpenAuthorization with an ID that which will be used to identify them via binding that OpenAuthorization ID to a user ID provided with our service.

Email and Password Account Storage

Users that sign up with the standard account creation method of email and password will have their account's user ID attached to their email. Their hashed passwords will be stored alongside this information.

Both OpenAuthorization and regular sign ups will have their accounts creation date stored.

User IDs	**OAuth ID**	**Usernames**	**Email**	**Hashed Password**	**Member Since**
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]
[User_ID]	[Oauth_ID]	[Username]	[Email]	[Hashed_Pass]	[DD/MM/YY]

General Database Plans

User data is, of course, not the only type of data that needs storing. On the app's server side, we would need to be able to store multiple datapoints across several different areas in order to provide the users with all of the relevant information to let the app even function at all.

Initial Research

The first stage of the process was to identify the key areas that would require databasing in the first place. This is a step that necessarily gets pushed towards the later stages of development a little, as it is much easier to fit a database around a properly solidified application framework than it is for one to find themselves needing to expand or completely refactor the database in response to changing requirements(Pocket App, 2019).

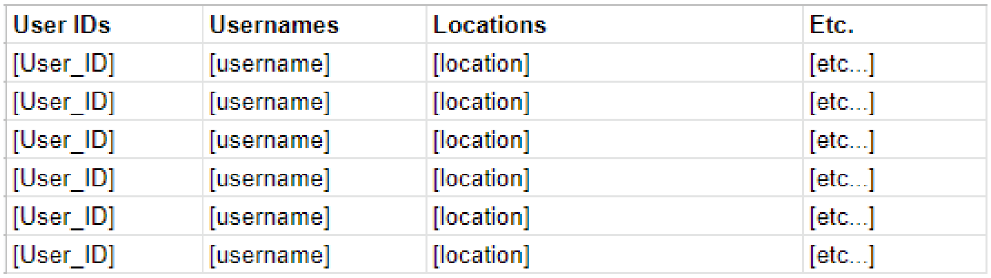
The earliest identified points of interest were:

- User Accounts
- Venue Information
- User-submitted data (such as reviews)

These areas were also partially determined by further research into database creation and management. Similar projects were found to use a table for the basic user information, one for each venue's information, and then a network of multiple tables tied into these to store information about comments, tags, likes, and so on.

Database Structure

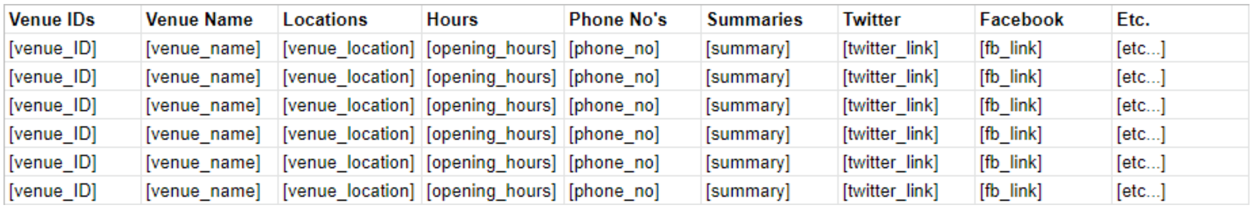
At its core, the database is a linked series of tables, each with referenced entries. For example, the user information table contains each user's unique ID - this is a string generated newly for each account upon creation and remains tied to it everywhere else in the application.



User IDs	Usernames	Locations	Etc.
[User_ID]	[username]	[location]	[etc...]
[User_ID]	[username]	[location]	[etc...]
[User_ID]	[username]	[location]	[etc...]
[User_ID]	[username]	[location]	[etc...]
[User_ID]	[username]	[location]	[etc...]
[User_ID]	[username]	[location]	[etc...]

It then also holds smaller pieces of data that the user can submit on their profile page. This is things like a display username, possibly their set location (which the mapping function could then use if the user doesn't have location data turned on), small text fields with their interests, or an "about me" section, and so on.

It is a similar story when it comes to the venue information, albeit with more detailed extra columns:

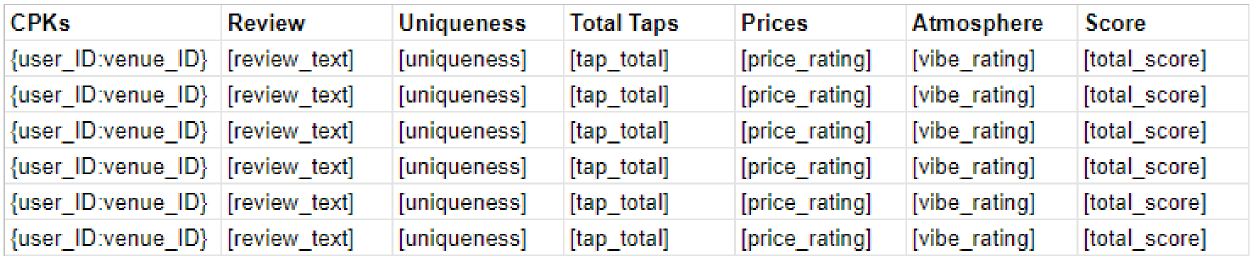


Venue IDs	Venue Name	Locations	Hours	Phone No's	Summaries	Twitter	Facebook	Etc.
[venue_ID]	[venue_name]	[venue_location]	[opening_hours]	[phone_no]	[summary]	[twitter_link]	[fb_link]	[etc...]
[venue_ID]	[venue_name]	[venue_location]	[opening_hours]	[phone_no]	[summary]	[twitter_link]	[fb_link]	[etc...]
[venue_ID]	[venue_name]	[venue_location]	[opening_hours]	[phone_no]	[summary]	[twitter_link]	[fb_link]	[etc...]
[venue_ID]	[venue_name]	[venue_location]	[opening_hours]	[phone_no]	[summary]	[twitter_link]	[fb_link]	[etc...]
[venue_ID]	[venue_name]	[venue_location]	[opening_hours]	[phone_no]	[summary]	[twitter_link]	[fb_link]	[etc...]
[venue_ID]	[venue_name]	[venue_location]	[opening_hours]	[phone_no]	[summary]	[twitter_link]	[fb_link]	[etc...]

Once again, each venue has a uniquely-generated ID. After that, this table can store the various set-in-stone elements of each venue - their contact information, socials, location, opening hours, and a short summary (possibly pulled from Google Maps API or user-submitted/reviewed).

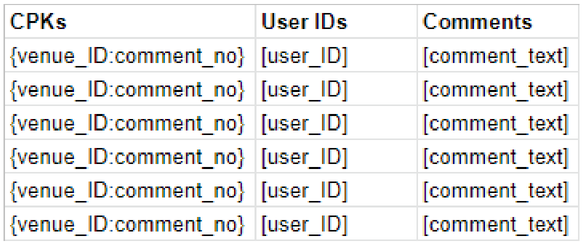
Anytime something in the app needs to display relevant information about either users or venues, it sends a quick lookup request through to the database, uses the ID of the item to find it in the table, and then pulls out whichever datapoint is needed to display.

When an table item's ID is used like this, it's known as a **Primary Key**. The primary key is what keeps that particular entry identified as a point on that particular table. This then feeds forward into how things like reviews and comments can be stored.



CPKs	Review	Uniqueness	Total Taps	Prices	Atmosphere	Score
{user_ID:venue_ID}	[review_text]	[uniqueness]	[tap_total]	[price_rating]	[vibe_rating]	[total_score]
{user_ID:venue_ID}	[review_text]	[uniqueness]	[tap_total]	[price_rating]	[vibe_rating]	[total_score]
{user_ID:venue_ID}	[review_text]	[uniqueness]	[tap_total]	[price_rating]	[vibe_rating]	[total_score]
{user_ID:venue_ID}	[review_text]	[uniqueness]	[tap_total]	[price_rating]	[vibe_rating]	[total_score]
{user_ID:venue_ID}	[review_text]	[uniqueness]	[tap_total]	[price_rating]	[vibe_rating]	[total_score]
{user_ID:venue_ID}	[review_text]	[uniqueness]	[tap_total]	[price_rating]	[vibe_rating]	[total_score]

This is how a table of user reviews would then look. Instead of its own generated ID as a primary key, a review uses a linked pair of the user ID leaving the review, and the venue ID that's receiving the review. These are known as "foreign keys" (as they are taken from a different table's primary keys), and the result is called a **Composite Primary Key**. The two immediate benefits to the Composite Primary Key system are firstly that having a user or venue ID as part of each CPK means that any other place using those IDs (such as a user or venue profile) can then display all reviews containing the relevant key, so visiting a user's profile can show all the reviews they have left, for example.



CPKs	User IDs	Comments
{venue_ID:comment_no}	[user_ID]	[comment_text]
{venue_ID:comment_no}	[user_ID]	[comment_text]
{venue_ID:comment_no}	[user_ID]	[comment_text]
{venue_ID:comment_no}	[user_ID]	[comment_text]
{venue_ID:comment_no}	[user_ID]	[comment_text]
{venue_ID:comment_no}	[user_ID]	[comment_text]

The second benefit is that each user/venue ID pair is treated as a unique key that can only exist in a single instance. Both to streamline the app and prevent "review bombing", a user cannot review a venue multiple times, only edit their previous review. This review entry in the table stores the various components, being the 5 categories as well as a text summary, and again is called by simple lookup requests.

The last example here is how a table would look for comments, or any similar "repeatable" entity that - unlike a review - may be left multiple times by a single user. In this instance, the CPK only takes in the Venue ID as a foreign key (the only part of it that can exist a single time), and pairs it with a generated comment ID primary key instead. In addition, the commenter's ID is stored but not in a unique referential format, so the same user ID can appear multiple times if needed.

This key-structuring technique is what forms the entire backbone of the database and can be extended to new tables with relative ease, while also providing a method for the rest of the app to communicate with it and quickly pull out relevant data with haste.

This diagram, by Branko Dimitrijevic, is a rudimentary outline of how similar social apps may structure themselves. The original source also contains another excellent introduction on how to create and structure databases in the style that our project largely draws from.

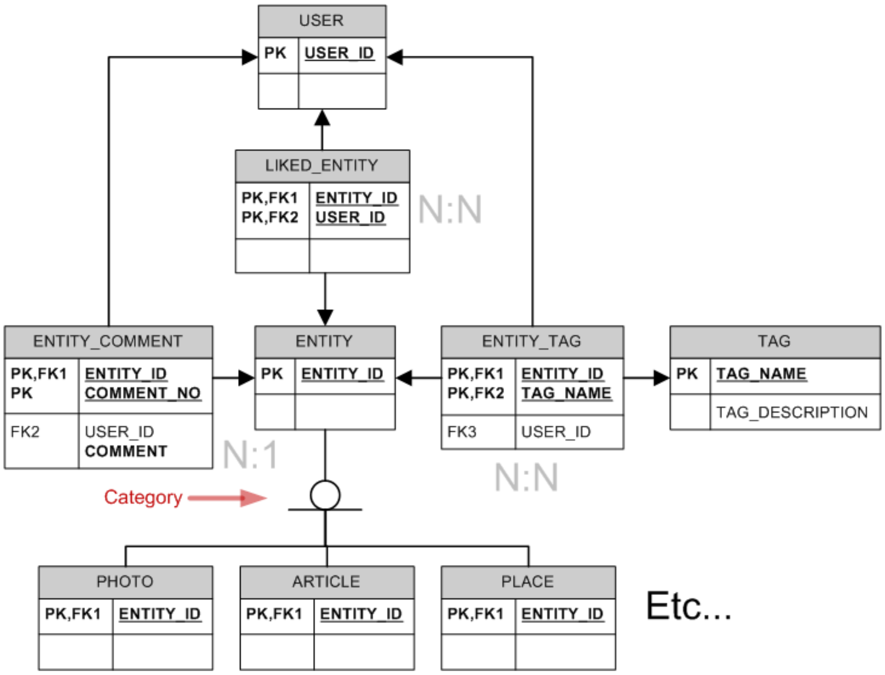


Image Credit: Branko Dimitrijevic, stackoverflow.com

Beyond the Scope/Chat Features

Moving forward, the app will start to include more social features such as, crucially, a chat/messaging service in order to connect with like-minded drinkers, creating a more satisfying experience for all involved. Tentative preliminary research into this area has yielded multiple different approach options for implementing such a feature, using tech stacks involving such things as MongoDB's NoSQL, or an extension on the current PostgreSQL implementation, Nginx, or possibly integrated with AWS to take off some of the server slack on our side (Kumar, 2022). From the storage half of the equation, depending on the level of security desired one can either store chat messages on the server in an NoSQL database or increase security by storing chat messages locally on user's devices, encrypting it before it is sent to the server and decrypted on the receiving device. Both of these options are worth considering for future social implementations.

User Experience and User Interface

UX (User experience design) and UI (User interface design) are two terms often used interchangeably and although the two are closely linked with some overlap, they are separate areas that have differences in goals and approach.

UX focuses on a user's journey through a product or application whereas UI focus on the aesthetics and visual elements of an application. The goal of a UX designer is to create an easy, efficient, relevant, and all-around pleasant experiences for the user (Lamprecht, 2022). UI designers work with the wireframe or structure provided by the UX designer and create a visually-appealing easy-to-understand interface (Career Foundry, 2021).

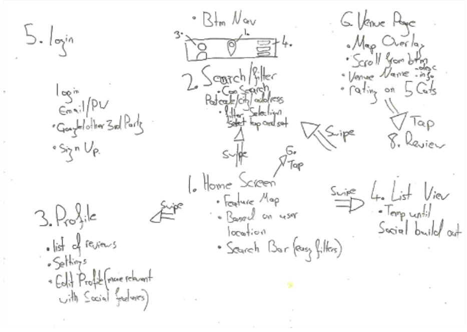
Initial Research

The first stage in development was choosing the tools and technologies to develop with. Research and comparisons of UI and UX development software led to a focus on Proto.io, InVision and Adobe XD due to their price, difficulty of use and time to master all being mid or low. After an hour of testing each piece of software, the decision was made to move forward with Adobe XD, primarily due to its integrations with other Adobe applications which weere already being used to create assets for The Pursuit of Hoppiness.

Before beginning on the wireframe for The Pursuit of Hoppiness, several tutorials were worked through to develop skills. Skills also significantly grew during the development process.

UX Planning

The goal of UX is to create an easy and efficient flow to allow users to complete their intended goals. The role can often be broken down into 5 phases: audience research, identification of users needs, generation of design ideas, prototyping and testing (Siang, 2021). We identified our target audience as craft beer drinkers and our research produced an audience profile of males primarily aged between 30 and 39 (Gjorgievska, 2021), with the need to find the right bar for them. This led to a more traditional design with an approach that minimized the clicks to view venue scores to one. Emphasis was also put on navigation within the app.

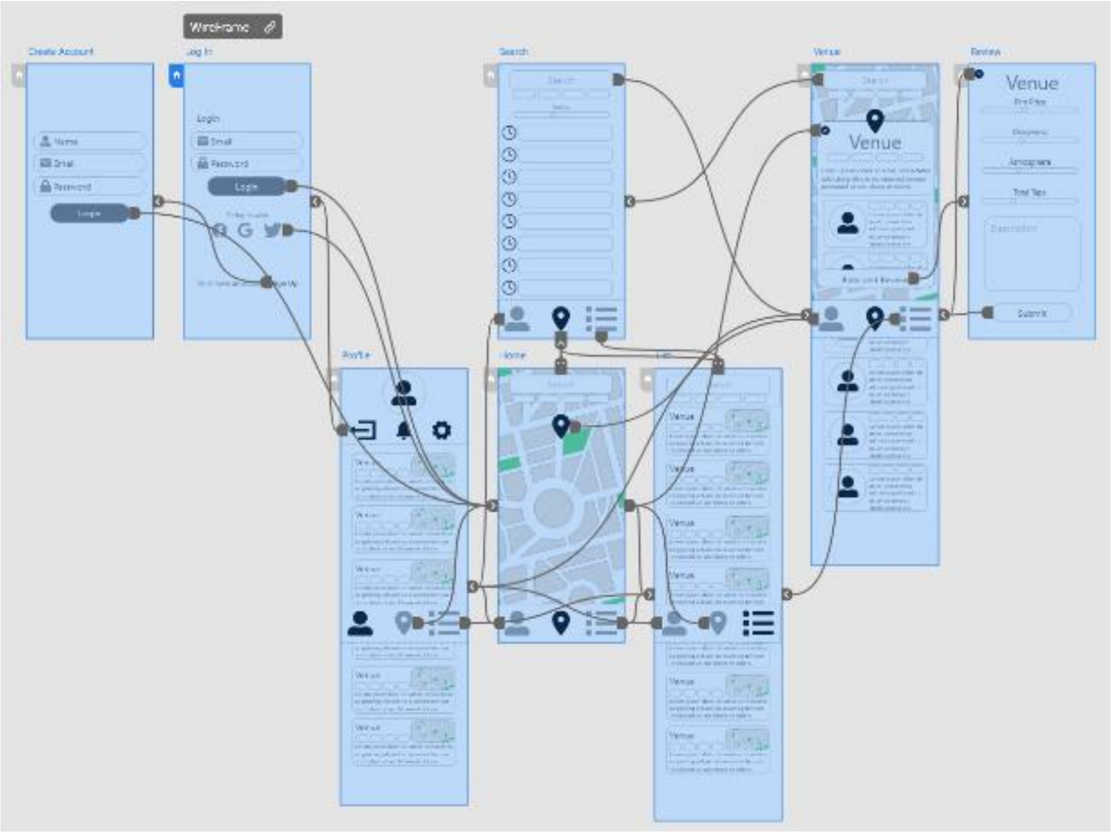


The home screen for existing users is the Map page. This screen initially centres on the user, giving them quick access to information on venues around them. It can be navigated similarly to other touch-based maps with pinching to zoom and swiping to move. Location can also be set using the search feature which has a text field for the search location and past searches. The List page presents venues based on distance. It can be filtered by our four review categories to raise/lower venues depending on preference. The account page allows users to view their past reviews, manage settings, and log out. When venues are clicked on the map or searched, they become centred on the map and give information on the venue as well as recent reviews.

Each page features a navigation panel that can be used to quickly move between the Profile, Map and List view.

UX Prototype

The wireframe was made using tools inside XD and the Icons 4 Design plugin. Development of the wireframe was also used as training in XD.



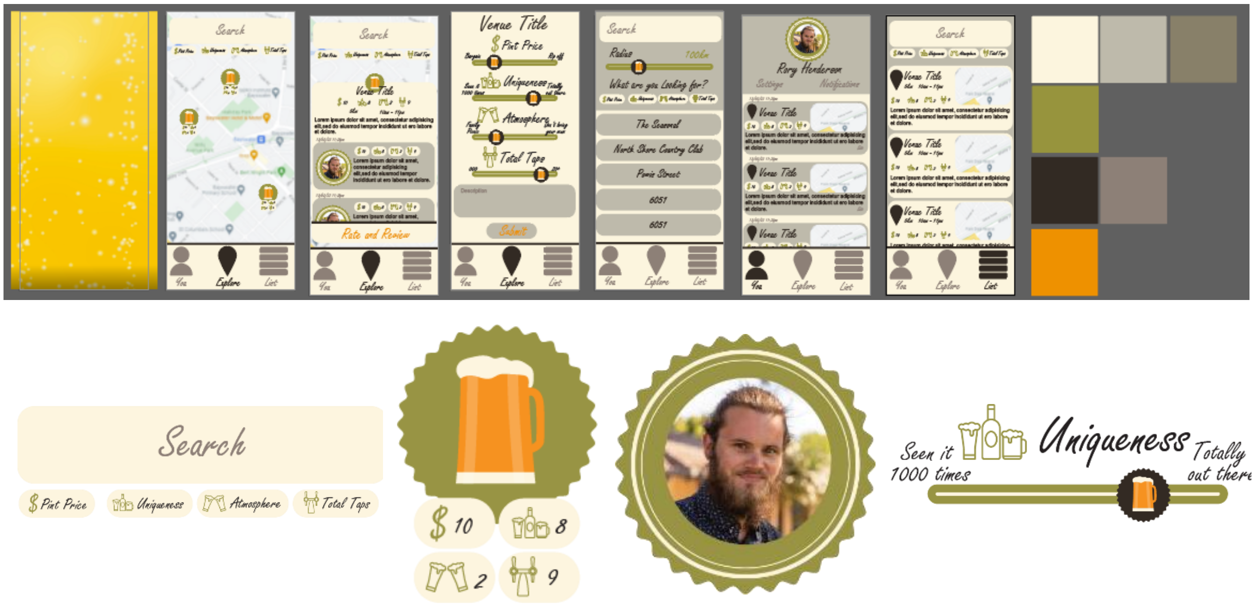
Please visit https://xd.adobe.com/view/707902ce-3eaf-48ed-b05c-ca30b6efacf0-252a/ for an active Prototype

During development of the wireframe, swipe navigation was removed as it wasn't supported by XD, and the Venue page was changed to a pop-up box map overlay instead of its own page.

UI Development

When developing the UI for The Pursuit of Hoppiness, emphasis was placed on consistency using already-established interface elements such as buttons and sliders as input controls, commonly used search field structures and map functionality, and use of already-established icons for common functions such as settings and notifications.

To keep the app consistent with existing branding, assets were reworked and expanded upon for use within the application. Some generic application symbols, such as the login, settings, notification and login icons, were taken from the Icons 4 Design plugin which allows non-attributed use of 5000 icons and symbols. These icons were colored to match the style guide.

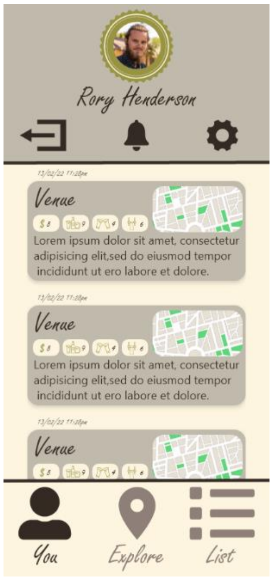


Initial Adobe Illustrator UI Asset Mock Up

Initial mock-ups of the UI were created in Adobe Illustrator and then imported into the wireframe XD file and worked to replace the wireframe icons. During this process changes were made to the UI and UX to improve functionality. This included moving the radius slider from Search to List and adding additional lines to create breaks between sections. Additionally, extra artboards were added to add filter functionality for the prototype.

Final Prototype

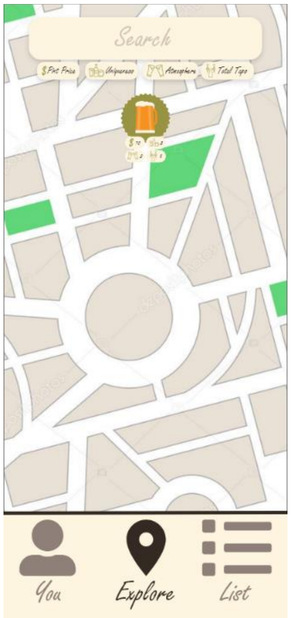
The current prototype demonstrates the UI and UX of the application well, with several limitations as a result of XD; limitations that will be addressed in future prototypes are editable text fields, active sliders, dynamic map and the addition of additional filters (currently only Tap Price is active).



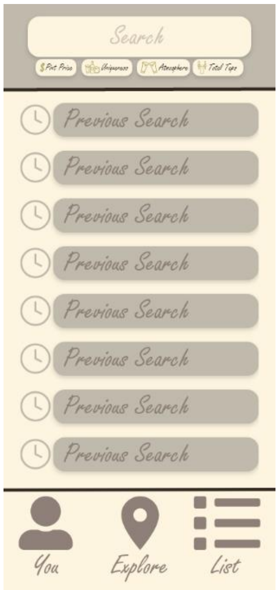
Lorem ipsum dolor sit amet, consectetur adipisicing elit,sed do eiusmod tempor incididunt ut ero labore et dolore.

Lorem ipsum dolor sit amet, consectetur adipisicing elit,sed do eiusmod tempor incididunt ut ero labore et dolore.

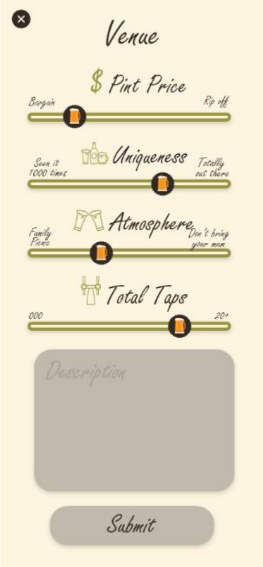
Profile Page



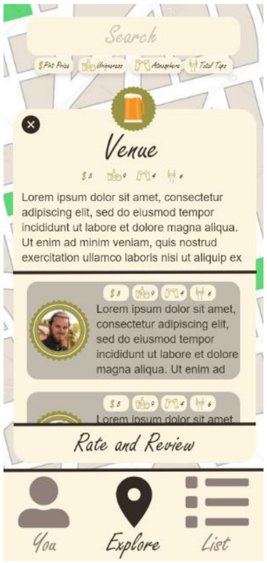
Home Page



Search Page



Review Page

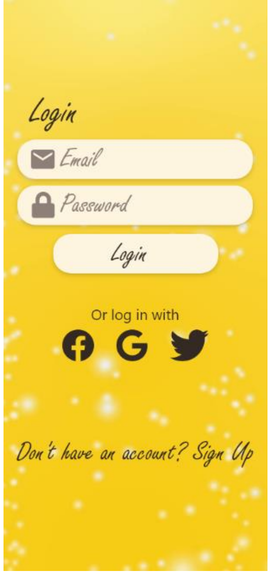


Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad

Lorem ipsum dolor sit amet,

Venue Page



Login Page

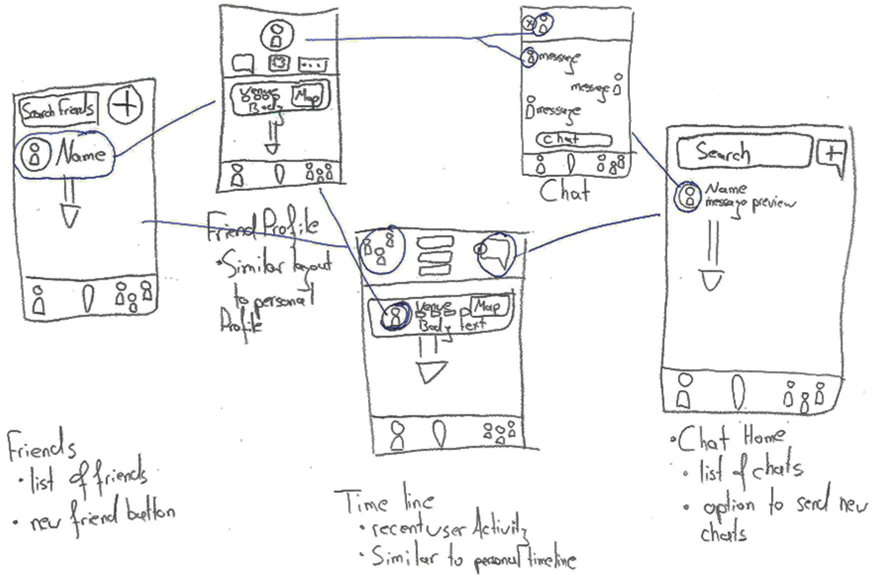
Please visit https://xd.adobe.com/view/1b06189c-b54c-4401-9739-23bed5b336fd-4e55/ for an active Prototype

Following feedback from the development team the bottom navigation bar will be moved up to allow room for phone navigation icons/buttons, phone battery, time and connection icons will be added to the top of the application, and the size and amount of text will be reviewed.

Next Stages

The current prototype was designed as per the scope for a minimum viable product and is missing major features that would significantly change the app UX and UI. The two biggest changes would come from the addition of Social Features and Venue Management Features.

Social features including friends, messaging and an activity timeline would be accessible through the navbar and would replace the current list button, which would be relocated as a smaller icon next to the search field. This feature would add an additional 5 major screens: Timeline, Friends, Friend Profile, Chat Home and Chat. The UI and layout of these screens will be consistent with the application in line with the style guide and keeping the bottom navigation. Navigation features specific to the social features will be located at the top of the page.



The venue management features will require a significantly different interface and will be a web portal rather than an app. Initially the features will be limited to the ability to edit their venue description and opening hours, reply to comments and view statistics on venue reviews. These features can be introduced with minimal changes in the app UI but future development including a venue timeline and the ability for patrons to make purchases in app will require a rework of the venue page.

References:

Career Foundry, 2021. UX Design vs UI Design - What's The Difference?. [video] Available at: https://www.youtube.com/watch?v=5CxXhyhT6Fc [Accessed 2 February 2022].

Gjorgievska, L., 2021. Australian Beer Market Statistics That Raise the Bar. [online] Take a Tumble. Available at: https://takeatumble.com.au/insights/lifestyle/beer-market-statistics/ [Accessed 2 February 2022].

Lamprecht, E., 2022. UX vs. UI Design: What's the Difference? [2022 Guide]. [online] CareerFoundry. Available at: https://careerfoundry.com/en/blog/ux-design/the-difference-between-ux-and-ui-design-a-laymans-guide/ [Accessed 1 February 2022].

Siang, T., 2021. 7 UX Deliverables: What will I be making as a UX designer?. [online] The Interaction Design Foundation. Available at: https://www.interaction-design.org/literature/article/7-ux-deliverables-what-will-i-be-making-as-a-ux-designer [Accessed 3 February 2022].

Ceci, L., 2022. Top U.S. mapping apps by downloads 2021 | Statista. [online] Statista. Available at: https://www.statista.com/statistics/865413/most-popular-us-mapping-apps-ranked-by-audience/ [Accessed 19 February 2022].

Google (n.d.). Google Maps Platform. [online] Google Developers. Available at: https://developers.google.com/maps.

Goteleport.com, 2022. What is OAuth | How OAuth 2.0 Works. [online] Available at: https://goteleport.com/blog/how-oauth-authentication-works/ [Accessed 13 February 2022].

Bisso, I., 2021. How to Model Inheritance in a Database. [online] Vertabelo.com. Available at: https://www.vertabelo.com/blog/inheritance-in-database/

Pocket App, 2019. The 4 stages of app development. [online] BusinessOfApps. Available at: https://www.businessofapps.com/insights/stages-of-app-development/

Kumar, P,. 2022. How To Build Your Own Custom Chat Platform: The Full Guide. [online] Contus.com. Available at: https://www.businessofapps.com/insights/stages-of-app-development/

Borelli, P,. 2018. NoSQL: Building Your First Application & Connecting To MongoDB. [online] CodeWall. Available at: https://www.codewall.co.uk/nosql-building-your-first-application-connecting-to-mongodb/

Roles

In our initial development period, we have assigned team members to focus on specific functions of the application to maximize our development timeline. The roles were created based on a standard development team structure (App Development Timeline: How Long Does It Take? - Existek Blog, 2019) and then divided based on interest once we broke down the requirements of our Scope.

Rory - Project Manager and UI/UX Designer

Rory proposed the idea of the Pursuit of Hoppiness and has taken on the role of Project Manager. In this role he communicates the project's vision to the rest of the development team, manages work assignments and team meetings and manages the project's timeline. Rory has some management experience but limited to non-technology-based projects.

He has also taken on the role of UI/UX designer managing the branding, look, feel and flow of the app. He has experience with graphic design primarily in relation to video graphics and is familiar with the Adobe Creative Cloud suite.

Gavin - Backend Product Manager and Developer

Gavin has some experience in software and database programming as well as management of both technical and non-technical stakeholders. Gavin's role while contributing to development will be to manage the expectations of stakeholders with respect to scope creep, budget, and timelines.

As this role is largely backend-focused, it is also important to act to provide good service to the front end teams while maintaining security and performance in line with the vision of the project.

Kyle - Backend Developer

Kyle is one of our backend developers, ensuring that the code enabling the application for the Pursuit of Hoppiness functions properly and enables our users to use the application fluidly and without error. He works together with the rest of the backend developers to improve the application and our users experiences while using the application.

Kyle contributes to the development of the application via things such as the integration of third-party API account creation and logins, password hashing, and the storage of new users' data and account details - such as passwords, their email, and account ID.

Evan - Backend Developer - Database Manager

Evan is one of the team members primarily working on backend solutions for databases and data storage generally. He is thus responsible for helping ensure that any user data submitted and stored is done so safely and securely, while communicating with the other developers to ensure that they can quickly access or store data as they need.

References:

Existek.com. 2019. App Development Timeline: How Long Does It Take? - Existek Blog. [online] Available at: https://existek.com/blog/app-development-timeline-how-long-does-it-take/ [Accessed 29 January 2022].

Scope & Limitations

Initial Scope for the Minimum Viable Product

There are 5 key criteria that are to be met for the product to be considered at the Minimum Viable Product Stage. For simplification this will be listed in dot point form.

Pursuit of Hoppiness must:

- Be available on iOS.
- Allow login using existing social media credentials. (Facebook, Google, etc.) and account creation in app.
- Allow a user to view a venue in either list or map form
- Allow a user to view basic venue information (opening hours, cost guide, beer selection)
- Allow a user to record a rating and a comment on an existing venue

These are the only goals for Phase 1. The goals are limited to reduce scope creep and keep development costs and time low. The timeline for completion and full integration of these goals is by week 15 of development.

Within these goals there are many sub-items that themselves will be quite in depth. The intention of this list is to provide a clear direction to the development team that these are the only main objectives that we should be working towards.

The scope for our first six weeks of research and development is to create siloed examples of:

- A UI/UX prototype with static data
- A map mockup with our chosen map provider with static data
- A clear development and storage strategy for user login
- A structure and plan for deployment of a database

Tools & Technologies

Google Maps

Gavin has had some experience using the Google Maps API in the past for small projects that needed to be both interactive and update dynamically.

Google Maps is chosen over Apple Maps even though initial deployment is planned for an iOS device. This is because Google Maps will provide an API for web-based, Android-based, and iOS-based applications which will allow for simplification of the product.

OpenAuthorization 2.0

OpenAuthorization's API was used to enable users to login through existing third-party services such as Google, Twitter, and Facebook. This is done so by OAuth 2.0 communication with these sites to retrieve an ID that is bound to a user ID within our application.

The reason for having chosen OAuth2.0 is due to the fact it is the industry standard that which the biggest tech sites utilize.

UI/UX

Development of the app's UI and UX was Rory's role. He has had design experience but has had no experience working with UI and UX development tools. After research, he settled on developing a prototype using Adobe XD. This was chosen due to its integration with the Adobe Creative Cloud Suite, which was already being used to develop design assets for the project. These Creative Cloud apps all required licensing through Adobe. The specific apps used were:

- Adobe XD (Version 47.0.22)
- Adobe Photoshop (Version 23.01)
- Adobe Illustrator (Version 26.01)
- Adobe After Effects (Version 22.01)

Additionally, Rory used online stock libraries to assist in asset creation, using premade vectors and customising them to fit with the app's style.

- Vecteezy - Unlimited subscription billed annually
- Storyblocks - Unlimited All-Access subscription billed annually

NoSQL Database

For putting together and maintaining a NoSQL database, Evan is building it on top of MongoDB, a flexible NoSQL databasing platform. To communicate between each other, the main app and the database need a middleman called a Web Server, and this will be primarily put together using Python 3.

For as much of the hardcoding as will possibly need doing, he will be writing it up using the Sublime Text editor, since it has both inbuilt support for most all commonly-used programming languages, and also natively interfaces directly with GitHub to streamline the process of editing commits and pushing them out to the live branch.

Testing

Early product testing is to be conducted in three stages, similar to how any IT product is developed (Thakar, 2019). Initially, in-house testing will be carried out to identify and rectify any major issues. It is intended that this will be ongoing throughout development. The second stage would be a "closed alpha" release to friends and family of the developers. This idea is to get a larger cross-section of users with different technical abilities that have not yet been exposed to the application. Some bias could creep into the closed alpha but it will still deliver good results. Following a successful closed alpha testing phase (of which there may be many), an open beta will be launched allowing for a larger pool of testers and a decreased amount of bias.

Unit testing for individual elements will be a requirement for a deliverable of any major section. For this reason a dedicated Quality Assurance developer will be employed to work alongside all developers to ensure that modules are robust and can fit within our Continuous Integration and Continuous Delivery plan.

References

Thakar, M. (2019). Different Stages of Game Testing. [online] KiwiQA. Available at: https://www.kiwiqa.com/different-stages-of-game-testing/.

Timeframe

We are developing the Pursuit of Hoppiness in an unconventional way compared to that of professionals; a product developed by a team of professionals would be developed sequentially (Tate, n.d.), however we find ourselves developing everything at once.

A sequential, standard development cycle from a professional team may look as such: first would come the analysis and planning stage, where the project is discussed and resources needed would be decided upon. Following this would be the UX design team would develop the structure and flow of the application follow by the UI design team, where designers work on the appearance of the application - things such as buttons, headers, logos are created. After this, the backend developers would work on the functionality of the program, programming things such as database integration, cloud functionality and other relevant areas. Finally, the frontend developers would then take all these elements; the UI, UX, and backend code, and sculpt it all together to mould everything into the first stage of a functional product.

Evan

Evan's main objectives for the app's development, as the primary database manager, will be to ensure that prior to the initial prototype deadline, at least a rudimentary database must be functional and able to neatly store all required user/venue data. After that, alongside the team's further app development, he will construct out the database to accommodate more varied entities such as comments, and the hosting of chat (if it's to be stored server-side).

Gavin

Gavin will focus on the Maps and coordinating activities between front end and back end development to ensure that they remain efficient and move together with each other. From this point of view, Gavin will be able to have an overview of pain points being experienced in all areas of the application and direct resources appropriately to deal with them. For example, when monitoring the closed alpha feedback and planning for the open beta release, a problem reported by a user needs to be accurately triaged and directed to the proper development team for a fix. A failure to do this may result in the symptoms being treated instead of the disease.

Prior to recruiting a dedicated resource for Quality Assurance, Gavin will be responsible for interim QA.

Rory

Rory's role in the development of the Pursuit of Hoppiness is the UX and UI designer. He will work towards hitting two primary deadlines, those being an active prototype in week 5 of development and a working app that is fully integrated with other services by week 15.

As there is a goal for an active prototype in Week 5 and UX development starting in week 3, the expected UX development cycle of 3-4 weeks and UI cycle of 2-3 weeks (App Development Timeline: How Long Does It Take? - Existek Blog, 2019) will be compacted into a total of three weeks, with tweaking and fine-tuning happening after the production of the prototype.

Kyle

Kyle's role in the development of the Pursuit of Hoppiness is that of a backend developer. He will work on the things behind the scenes, such as the sign-up and account storage process.

Andrew

Andrew was with us up to week three before unenrolling. He was initially assigned the role of Database Development, shared with Evan. His first two weeks of development followed the groups before he left.

Week	Evan	Gavin	Rory	Kyle
1	Discussion and development of the goals for the app prototype Discussion of current skills in the group and division of roles based on this Broad timeline set for project			
2	Research into database solutions Development of a 15 week timeline for his role	Research into maps Development of a 15 week timeline for his role	Research into UX and UI Consideration of applications available for prototyping Develop written outlines for UX Collate existing UI design assets Further development of graphic assets for the app Development of a 15 week timeline for his role	Research into third-party login services Research into self-managed users Development of a 15 week timeline for his role
3	Communication with UI/UX development team members to determine the scope of data to be stored	Start to develop a web-based mockup of maps solution	Develop a drawn UX outline Learn prototyping application Begin creating prototype using chosen application	OpenAuthorization integration research

4	Starting to draw up prototype/mockup blueprints for both the database style and database structure	Delivery of mockup of maps	Add design elements to the UI Begin drafting style guides Testing inside the team Shoot video content to produce brand video	Research storage of user ID/account number into database Shoot password hashing and 3rd party login portion of brand video
5	Delivery of fleshed out mockups and plans	Develop position descriptions	Static/interactive prototype of the app Style guide Brand asset library Produce brand video with app UI mockup	Create mockup database structure for account information storage
6	Begin building out a basic SQL framework and database environment Continue communications with the front-end team to analyse DB requirements	Refine map design elements	Make adjustments based on internal feedback to prototype Work with Gavin to refine map design elements to match current branding	Align database structure with Evan's database mockup
7	Collaborate on API planning, and start development proper on user info databases	Develop a plan for the core API for communication with the front end	Finish final UI adjustments Share required functionality with development team	Begin development of self-managed account creation

8	Complete development and delivery of core account database	Development of database to API integration Delivery of mock API to facilitate frontend development	Working with Kyle and Evan to integrate self-managed account signups	Integrate self-managed account signups
9	Primary database expansion Separate DBs for storing bar information as well as non-public data needed for management purposes	Development of API	Coordinate with backend team for closed alpha expectations	Research into hashing of users' passwords
10	Delivery of alpha backend Coordination with frontend team for closed alpha expectations	Delivery of alpha backend Coordination with frontend team for closed alpha expectations	Finish development of functional prototype for alpha testing	Integrate password hashing for self-managed users' accounts
11	General DB maintenance, ensuring the client-side is able to access and store data securely	Closed alpha briefing and delivery	Closed alpha briefing and delivery Development of further branded assets and marketing material pre–beta launch	Closed alpha briefing and delivery

12	Continued maintenance and management	Collation of closed alpha feedback	Collation of closed alpha feedback Revisions of UI and UX to reflect feedback	Bug squashing from closed alpha feedback
13	Continued maintenance and management	Collation of closed alpha feedback	Revisions of UI and UX to reflect feedback Development of early UI and UX for future features	Bug squashing from closed alpha feedback
14	Continued maintenance Begin preliminary exploration into future beta requirements Cloud migration	Close out of closed alpha Develop roadmap to open beta Cloud migration	Development of early UI and UX for future features Plan targeted marking strategy for open beta Cloud migration	Contribute to development roadmap to open beta Cloud migration
15	Minimum Viable Product delivery. Will be used for a soft launch and to approach investors to enable further development.			

References

Existek.com. 2019. App Development Timeline: How Long Does It Take? - Existek Blog. [online] Available at: https://existek.com/blog/app-development-timeline-how-long-does-it-take/ [Accessed 29 January 2022].

Tate, T., n.d. How Long Does a Mobile App Development Project Take? | Clutch.co. [online] Clutch.co. Available at: https://clutch.co/app-developers/resources/how-long-does-mobile-app-developmet-project-take [Accessed 2 February 20

Risks

Cost Overrun

Controlling costs in initial phases will allow us to commit more time to development activities. One key expense point for a small organisation could be cloud fees. These can be eliminated completely until they become a necessity. Utilising clever development of microservices on local machines before migrating the system to an IaaS provider will allow us to control costs significantly.

Security

API Key Storage

As the project relies on numerous external services such as Google Maps, we must take steps to keep our API keys safe. An API key released into the wild could result in huge costs if not detected early or if measures arent taken to restrict the usage of the key (Alp, 2022). Secret stores should be utilised to prevent any API keys being required to be embedded in code directly, and git pre-commit hooks should be utilised to prevent inadvertent publishing of secrets to repositories.

Password Storage

As an account is required to access and use the Pursuit of Hoppiness, there is always the risk of data breaches exposing user's information to wrongdoers. We plan to combat and prevent this in two ways; the first being that passwords be inherently strong. They must meet the requirement of eight characters in length, while also containing at least one uppercase letter and special character. The second action being the implementation of password hashing, password hashing being a process in which passwords are transformed (hashed) from their initial states into a new string of letters and numbers. When logging into the website again, the password you enter is then hashed so as to match your intial password, compared to the hashed string in our database, and then approved or denied depending on whether the password is correct or not.

Complexity

Controlling complexity will be important. An overly complicated system will drive up costs by increasing security risk factors, development time, and technical debt. We aim to control complexity by limiting development to small achievable goals and limiting as much scope creep as possible. It may be prudent in some cases to extend development time to ensure that shortcuts are not taken that will lead to an increase in complexity of the system.

Poor User Adoption

An app without users cannot be successful and this is particularly true for an app that’s core functionality is built on user contributions. To mitigate this issue initial data seeding will be outsourced to a market research firm to have user generated content available from launch day. Additionally, the UI and UX team has scheduled time to work on Branding and marketing material for the beta launch, which although it isn’t in their traditional scope, is assigned to them due to our small organization size.

Designing unnecessary features

A major step in the UX development timeline that was skipped in this project is Research and Understanding the intended audience. Although we created a user profile of our intended audience, we were unable to perform in-depth analysis and traditional market research into this group, due to limited online research and no budget to conduct research. Planned features such as the social expansion may not be of interest to our users and would be a significant waste of development resources. This issue is hard to mitigate pre-launch, but after an initial user base is established listening to feedback and actively pursuing it with questionnaires would help.

Ethical Risks

The Pursuit of Hoppiness is constructed as a social and fun community review app, however the app's focus around the consumption of alcohol introduces two ethical considerations when discussing the user base.

Age Limit

In researching similar existing services and websites that featured alcohol, it was noticed that multiple sites contained an age gate such as Gage Roads Brewing, which requires users to verify with a "yes" or "no", and Jim Beam which requires users to enter their date of birth. Researching further it was learned that this is not a legal requirement in America (Thomson, 2017) for websites featuring alcohol. We were unable to find any sources that discussed whether it was or was not a legal requirement in Australia, so presumably it isn’t, however potential new laws will require social media sites, which the Pursuit of Hoppiness may fall under, to take “All reasonable steps to verify” that users are over the age of 16 (Buckley, 2021).

“All reasonable steps to verify” is vague, however in 2019 there was an inquiry into age verification for online wagering and online pornography by the Australian Parliament which discussed multiple methods for online age verification. There were multiple suggestions for using existing government identification methods for this purpose including existing government-issued ID documents or the Trusted Digital Identity Framework, however these forms of identification also verify a user's identity which in our context is far more information than we need. There were also suggestions to use publicly-available consumer information and databases, such as the Electoral Roll (which is something that Equifax said they do), which would only include individuals aged 18 and above, however again this introduces the issue of verifying a user's identity. Another solution presented was using a service such as Yoti Age Scan which can identify a person's age by looking at their face in around 1 to 1 ½ seconds with an error rate of 0.31% when the threshold is set to 25 (Standing Committee on Social Policy and Legal Affairs, 2019).

As there is currently no legal requirement to verify a user's age, these methods are currently not required.

Asking for a user's date of birth also presents an issue as it classifies as "personally identifiable information" and there have been cases of apps being blocked from the Apple App Store for asking for this information (Age Restrictions on Apps | Apple Developer Forums, 2016).

Despite there currently not being any legal requirement to do so, having users verify their age seems to be best practice. With this information in mind, when creating a profile in our app users will be required to select yes or no to the prompt are you over 18. If the user selects no they will be directed to the DrinkWise Landing Page which can direct underage users to more resources.

Alcohol Consumption

Pursuit of Hoppiness focuses on the craft market and social aspects of drinking, however it would be irresponsible to ignore the potential health risks to creating an app that encourages and promotes the consumption of alcohol.

Short term health risks of alcohol consumption include injuries, drowning, motor vehicle crashes, violence, alcohol poisoning and risky sexual behavior, and long-term risks include increased risk of cancer, high blood pressure, weakening of the immune system and social issues (Drinking too much alcohol can harm your health. Learn the facts | CDC, 2021). To mitigate these risks, the Australian Government recommends no more than 4 standard drinks in one day and no more than 10 standard drinks in a week (Managing your alcohol intake, 2020).

The addition of common safe drinking slogans, such as “Drink Responsibly” and “Know Your Limit” can be scattered around the app to help provide subtle reminders to users. Additionally, linking to resources such as DrinkWise in the main menu can provide awareness and support.

To further support safe drinking behaviors, the app could feature low and non-alcoholic beverages which is a quickly growing industry. In 2020 the share of non- and low-alcohol beer, wine and spirits incandesced to 3% of the total beverage market with an expected 31% growth of the non-alcoholic industry by 2024 (IWSR Drinks Market Analysis, 2021). The sales of non-alcoholic drinks at BWS and Dan Murphy's have almost doubled over the last 12 months (Brescia, 2021). Promoting these beers encourages their consumption and helps reconstruct the current social norms that result in 35% of 18-44 year olds saying they would hide low or lower-strength alcohol products while drinking socially (Zero, low and mid-strength alcohol – Australian consumers’ prevalence, practices and attitudes, 2021).

References

Brescia, P., 2021. Data shows the non-alcoholic drinks sector is booming. [online] SmartCompany. Available at: https://www.smartcompany.com.au/industries/retail/non-alcoholic-drinks-growth-australia/ [Accessed 13 January 2022].

Buckley, J., 2021. A new privacy code could compel social media platforms to take age verification seriously. [online] Business Insider Australia. Available at: https://www.businessinsider.com.au/online-privacy-code-australia [Accessed 10 January 2022].

Cdc.gov. 2021. Drinking too much alcohol can harm your health. Learn the facts | CDC. [online] Available at: https://www.cdc.gov/alcohol/fact-sheets/alcohol-use.htm [Accessed 13 January 2022].

Developer.apple.com. 2016. Age Restrictions on Apps | Apple Developer Forums. [online] Available at: https://developer.apple.com/forums/thread/20821 [Accessed 10 January 2022].

DrinkWise Australia. 2021. Zero, low and mid-strength alcohol – Australian consumers' prevalence, practices and attitudes. [online] Available at: https://drinkwise.org.au/uncategorized/zero-low-and-mid-strength-alcohol-australian-consumers-prevalence-practices-and-attitudes/# [Accessed 13 January 2022].

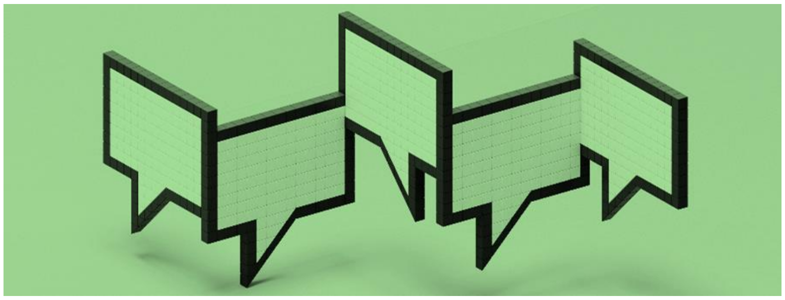
Healthdirect.gov.au. 2020. Managing your alcohol intake. [online] Available at: https://www.healthdirect.gov.au/managing-your-alcohol-intake [Accessed 12 January 2022].

IWSR Drinks Market Analysis, 2021. No- and Low-Alcohol Products Gain Share Within Total Beverage Alcohol. [online] Available at: https://www.theiwsr.com/wp-content/uploads/IWSR_No-and-Low-Alcohol-Gains-Share-Within-Total-Beverage-Alcohol-2021.pdf>[Accessed 12 January 2022].

Standing Committee on Social Policy and Legal Affairs, 2019. Inquiry into age verification for online wagering and online pornography. Canberra: Parliament of australia, p.Chaper 2.

Thomson, J., 2017. The Surprising Truth About Why Alcohol Websites Make You Enter Your Age. [online] HuffPost. Available at: https://www.huffpost.com/entry/liquor-website-age-verification_n_59c3b549e4b06f93538cdd18/amp?fbclid=IwAR2n_Ua4ppJzEGM3jrviGFRadP4K-1er1EF2Z23NCM4tG2M7umgVz06eO1w [Accessed 10 January 2022].

Alp, O., 2022. Unauthorized Google Maps API Key Usage Cases, and Why You Need to Care. [online] Medium. Available at: https://ozguralp.medium.com/unauthorized-google-maps-api-key-usage-cases-and-why-you-need-to-care-1ccb28bf21e [Accessed 19 February 2022].



Group processes & communications

Our team is scattered across Australia working in different time zones with different work, life and school commitments. To best facilitate communication and collaboration we use a variety of cloud-based synchronous and asynchronous tools.

Microsoft Teams

As no member of the current team lives within 700 km of the other, it is important that we use some online tools to enable collaboration

For general collaboration the Office 365 suite is selected as it is an industry standard and offers more interoperability with external vendors than the offerings from the nearest competitor, Google.

Microsoft Teams is the backbone of our communication. We have a weekly recurring full-team meeting where each member presents updates on their progress, discusses agenda items, and plans for future developments. From these meetings we can schedule additional calls as required as a full team or in smaller breakout groups. These form our primary synchronous communication.

During meetings we apply proven team communication strategies including: being inclusive, hearing from all members of the team, being clear and showing respect (Alexandra, 2016).

If a group member is unable to attend a meeting they respond to the invitation or Teams discussion with a reason ahead of the meeting. The meetings, agenda items and actions are recorded from these sessions, so members are still able to track progress and continue developing.

Teams also supports asynchronous communications though channel posts, allowing users to post new discussion items, or comment on an existing discussion as they are able.

Other tools built into Teams, such as SharePoint and lists, are also used regularly to support collaborative work and organization.

References

Alexandra, M., 2016. 6 proven project team communication strategies. [online] CIO. Available at: https://www.cio.com/article/238521/6-proven-project-team-communication-strategies-2.html [Accessed 12 January 2022].

Skills & Jobs

After consideration we choose that the roles that would most benefit our development are a Mobile Developer, Web Developer a Quality Assurance Engineer/Quality Assurance Automation Engineer and a Cloud Specialist.

Mobile Developer

About Us

The Pursuit of Hoppiness is a community-sourced bar rating and review app that can be used to filter by specific categories to allow beer enthusiasts to broaden their beer horizons, beer bargain hunters to find and deal, and all beer lovers to come together in one place. It is the “Pursuit of Hoppiness”. We are a new and quickly growing team that is fast paced, fun to work with and always up for a pint. The ideal candidate will be similarly driven, able to work autonomously, detail-oriented, keen to set a direction on technology and take ownership of the delivered results, have excellent communication, presentation, and collaboration skills.

Role Description

We are seeking to hire a mobile app developer. Ideally cross trained in iOS and Android operating system application development. You would work on anything encompassing mobile development by using your expertise in this field to take on the developmental duties such as the expanding, tweaking, and upgrading of our mobile applications infrastructure for both iOS and Android systems. We would look for a self-driven employee who excels in working as a one man unit but possesses the skills to later work in a team environment should we expand. As this is potentially a 100% remote position, great communication skills are required.

Generic Skills

- Great communication
- Love of beer

Specific skills and experience

- 2+ Years in Mobile App Development
- Mobile app development. iOS preferred, both iOS and Android advantageous
- Version control systems. Git Preferred
- Good understanding of OO programming and design patterns
- Microsoft Office 365 suite
- Familiarity with REST APIs
- Familiarity with Google APIs, specifically Google Maps.

Qualifications

- Computer Science or other IT related degree or relevant experience

Web Developer

About Us

The Pursuit of Hoppiness is a community-sourced bar rating and review app that can be used to filter by specific categories to allow beer enthusiasts to broaden their beer horizons, beer bargain hunters to find and deal, and all beer lovers to come together in one place. It is the “Pursuit of Hoppiness”. We are a new and quickly growing team that is fast paced, fun to work with and always up for a pint. The ideal candidate will be similarly driven, able to work autonomously, detail-oriented, keen to set a direction on technology and take ownership of the delivered results, have excellent communication, presentation, and collaboration skills.

Role Description

We are searching for a Web Developer to design our web based front end for the Pursuit of Hoppiness application. The Pursuit of Hoppiness application requires a web based front end for administration tasks and for access to our Business Insights product offering. This role is 100% remote and suits a self motivated individual who can work autonomously but can slot into a larger team whenever required.

Generic Skills

- Great Communication skills
- A natural collaborator and effective team player
- Strategic thinker
- Love of beer

Specific skills and experience

- 2+ Years in Web Development
- Experience with Javascript and any Frameworks
- Strong understanding of Bootstrap, HTML, CSS
- Experience with Rest APIs
- Familiarity with Google APIs, specifically Google Maps.
- Version control systems. Git Preferred

Qualifications

- Diploma level or greater in a relevant field or relevant experience

Dedicated Quality Assurance Engineer/Quality Assurance Automation Engineer

About Us

The Pursuit of Hoppiness is a community-sourced bar rating and review app that can be used to filter by specific categories to allow beer enthusiasts to broaden their beer horizons, beer bargain hunters to find and deal, and all beer lovers to come together in one place. It is the “Pursuit of Hoppiness”. We are a new and quickly growing team that is fast paced, fun to work with and always up for a pint. The ideal candidate will be similarly driven, able to work autonomously, detail-oriented, keen to set a direction on technology and take ownership of the delivered results, have excellent communication, presentation, and collaboration skills.

Role Description

The Quality Assurance engineer ensures our software meets our standards for security, reliability, and usability. The succesful applicant will operate largely autonomously and will need to be self motivated to achieve great results. The successful applicant will also possess great communication skills with all levels of employees along with a passion for improvement. You will also be required to deliver plans for testing and the results of each test cycle to both technical and non-technical staff.

Generic Skills

- Communication skills
- A passion for problem solving
- Experience with automated testing tools
- Experience developing testing plans
- Experience working with technical and non-technical partners, with the capability to influence
- Ability to write unit tests
- Love of beer

Specific skills and experience

- 5+ years experience in a programming related field or 2+ years in a QA field
- Experience with the O365 suite of applications.
- Python
- PostgresQL
- Google APIs
- iOS
- Flask
- REST APIs

Qualifications

- Computer Science or other IT related degree

Cloud Specialist

About Us

The Pursuit of Hoppiness is a community-sourced bar rating and review app that can be used to filter by specific categories to allow beer enthusiasts to broaden their beer horizons, beer bargain hunters to find and deal, and all beer lovers to come together in one place. It is the “Pursuit of Hoppiness”. We are a new and quickly growing team that is fast paced, fun to work with and always up for a pint. The ideal candidate will be similarly driven, able to work autonomously, detail-oriented, keen to set a direction on technology and take ownership of the delivered results, have excellent communication, presentation, and collaboration skills.

Role Description

We are seeking a passionate cloud specialist to conduct a migration and ongoing development of our cloud infrastructure for Pursuit of Hoppiness. The role will require you to be the go-to person for cloud related activities and assist development teams in creating applications with a cloud first strategy. The successful applicant will be a great team player and ideally a lover of good beers.

Generic Skills

- Communication skills
- A natural collaborator and effective team player
- Strategic thinker
- Good documentation Skills
- Love of beer

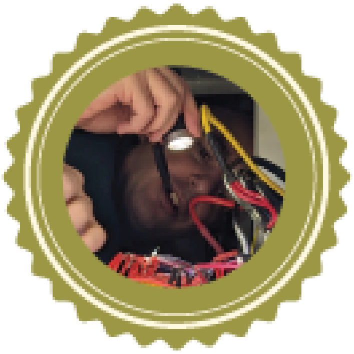
Specific skills and experience

- 2+ Years in Cloud Support
- Google Cloud Platform experience. Azure and AWS a positive
- Experience with the O365 suite of applications.
- One or more programming languages. Not one specifically, but the ability to learn one is important
- Docker and Kubernetes proficiency
- Experience with Linux Systems
- Experience with CI/CD tools

Qualifications

- Computer Science or other IT related degree

Reflection



Gavin Jamieson
S3925654

I feel that teams perform better when there is a social aspect to the group instead of just a number of tasks that have to be performed for a common outcome. This is hard to achieve in such a short time frame and even harder to achieve across a country and three time zones. I feel that this could be an area for improvement however with every team member having other responsibilities outside of this project, not a lot of time can be devoted to team building.

Despite the lack of team comradery I was surprised at how much could be accomplished when tasks were broken down into small enough manageable pieces. I am also further surprised that the team was able to absorb the increased workload so easily after losing 20% of the available labour.

I have learned that while groups can function with a common goal, another way to drive success is with a defined leader. Rory assumed this role in the group and I feel this contributed significantly to its success. Good job.

Rory Henderson
S3933584

We maintained the momentum from A2 and continued to work productively as a group. Reading over the reflection from A2 and the Feeback from the Assessment it is good to see that the changes we proposed have been implemented at the group have benefited as a result.

During this task I took on the role of Project Manager and managed the division of tasks week to week. As a result, I took on additional roles and tasks that in reflection could have been delegated more. I need to have more trust in my team and make more use of their skill sets.

I would also have been beneficial to have reached further functionality of the application during our development window. We reached our Scope but could have aimed higher as a team.

I am honestly surprised and grateful to have the whole team working and supporting the development of what was initially my idea.

My jobs and past experiences have had me working in a range of groups with different goals. This, however, was the first time I have worked fully remote on a project. Reflecting on how we used Teams and generally communicated you can see learning a growth from all team members in this area.

I had also never worked developing an app before, so my knowledge in the app development cycle has grown considerably.

Evan Phelps
S3934101

After the very diverse learning experience that was Assessment 2, Assessments 3 & 5 have been comparatively much smoother. Despite not only the increased workload of the more difficult and multiple tasks, but also the early departure of a team member, we generally had a much better time making effective progress with the work.

One big takeaway from this (for me, at any rate) has to be the absolute importance of a solid chairman/leadership role, who can hold everything together and ensure that the group runs like a well-oiled machine. If it wasn't for Rory stepping up and doing such a bang-up job in that role, I can confidently say that I would have found it harder to keep pace with the rest of the group as I can personally find it very tricky to track what particular work is still undone or where people may need my help to fill something out.

As for improvements, I really have to say that either moving away from MS Teams, or fully committing to Teams by installing the desktop app would have been better. The website is very unwieldy and if you don't have the page actively open, it's easy to miss important activity. There was more than one occasion when someone sent a crucial message or question through to the project discussion page and it would just get left unread for way too long. Also, all the notifications going into my spam inbox certainly didn't help.

Kyle Ross
S3930665

Working together with the other members of Sharper Tech was similar to that of assignment two, a positive experience from the otherwise daunting idea of group work. The team did well to communicate what was expected of us within the assignments requirements and the timeframe in which they must be completed in. This group work has taught me that having a set deadline as well as a team that relies upon me completing my work is when I am most motivated and focused on my work, I'll admit that often in this project I was only able to continue working when faced with the knowledge that the other members were depending on me to pull my weight else a heavier workload would fall upon them.

Though it shouldn't have surprised me having him shown his leadership skills in our previous assignment, I was still blown away Rory's seemingly never-ending work ethic and his ability to coordinate the group with who took on what work and when it was required. Gavin's ability to perform more complex tasks I personally would not have been able perform was a pleasant surprise, and Evan's willingness to propose new ideas and question others was surprising and opened up conversations and discussions which allowed us to improve.

If any one thing was to be improved, I'd personally recommend we'd have used another chatting service alongside Microsoft Teams, as outside of weekly video-calls there wasn't much an active thread of conversation between the group outside of direct messages. That being said, the majority of our problems were still able to be ironed out each week via the video calls.

Group reflection

What went well?

After A2, the group used our personal reflections and the feedback from our tutor on the assessment to make a few structural changes in order to improve the quality of outcome for A3. This included weekly recurring meetings being scheduled at the beginning of the development process, a single chair and the creation of a more structured group site that had all the required areas as heading and subheadings. This structure improved communication and allowed team members to track progress of all areas of the project easily.

Our team also developed their familiarity with Teams using its channels to break work down into specific projects, shared files and chat functions more efficiently in A3. The team also expanded their use of shared documents in Word, the use of OneDrive, and even used Stream for the A5 assignment

During A3 we unfortunately had one of our Team members unenroll from the course which took us down to a team of 4. We were able to recover from this loss and redistribute responsibilities around the Team, still allowing us to achieve our scope.

As our group reflection stated for A2 we continued to work well as a group with friendly, professional, and respectful communication practices that allowed team members to express ideas, ask questions, and learn from each other.

What could be improved?

As part of our scope, we intentionally assigned different areas of development to different team members without the goal of integrating these areas over the course of our 6 weeks of development. In hindsight we could have had the time to begin the integration process which would have improved the final deliverable and learning in the team.

A better integration of a group text-chat could have been developed, as messaging was often done peer-to-peer, often our talks and discussions as a whole group were only performed via our weekly meetings. This was a common thread in the individual feedback from all team members.

At least one thing that was surprising?

It was surprising to see the adoption of the Pursuit of Hoppiness idea between the group. Each member developed an understanding of the project idea and (hopefully) developed a sense of ownership of their area of research.

It was also surprising how satisfying it is to see the git commit logs being filled up with other team member’s work. With the work being 100% remote it gave assurance that there was work being done periodically and not just slapped together at the end.

At least one thing that you have learned about groups?

The added structure that came from moving the Teams meeting to weekly recurring meeting with a single chair reinforced the need for clear leadership in team project situations.

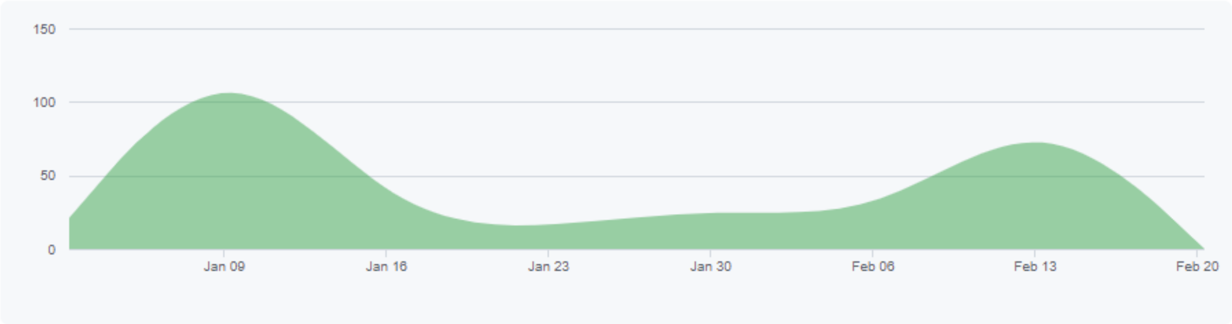
There was also an initial fear of fully online group collaboration, but through the completion of 3 group assessment tasks it is good to see that it is possible. Collectively, each team member learned something about the challenges of remote work with the added complication of working with people that they are unfamiliar with.

Github Insights

The Assessment 3 PDF asks us to reflect on how well the git audit log reflects on our team's contributions. It is our opinion that due to the different ways in which the team members contributed, there is no real metric from the logs that can be used. For example, we couldn't just use a simple count of commits as some team members committed large blocks of work and others made multiple small commits. We couldn't use the number of lines of code as some team members would modify dozens of lines in one go with minor changes when conducting reviews. Further to this, there was work carried out which was done outside of the web page by all the team members that contributed to the overall result.

To apply a practical example, a single team members contribution to this project well exceeded that of any other member of the group, however the metrics available on the github insights page can in no way be construed to reflect this fact.

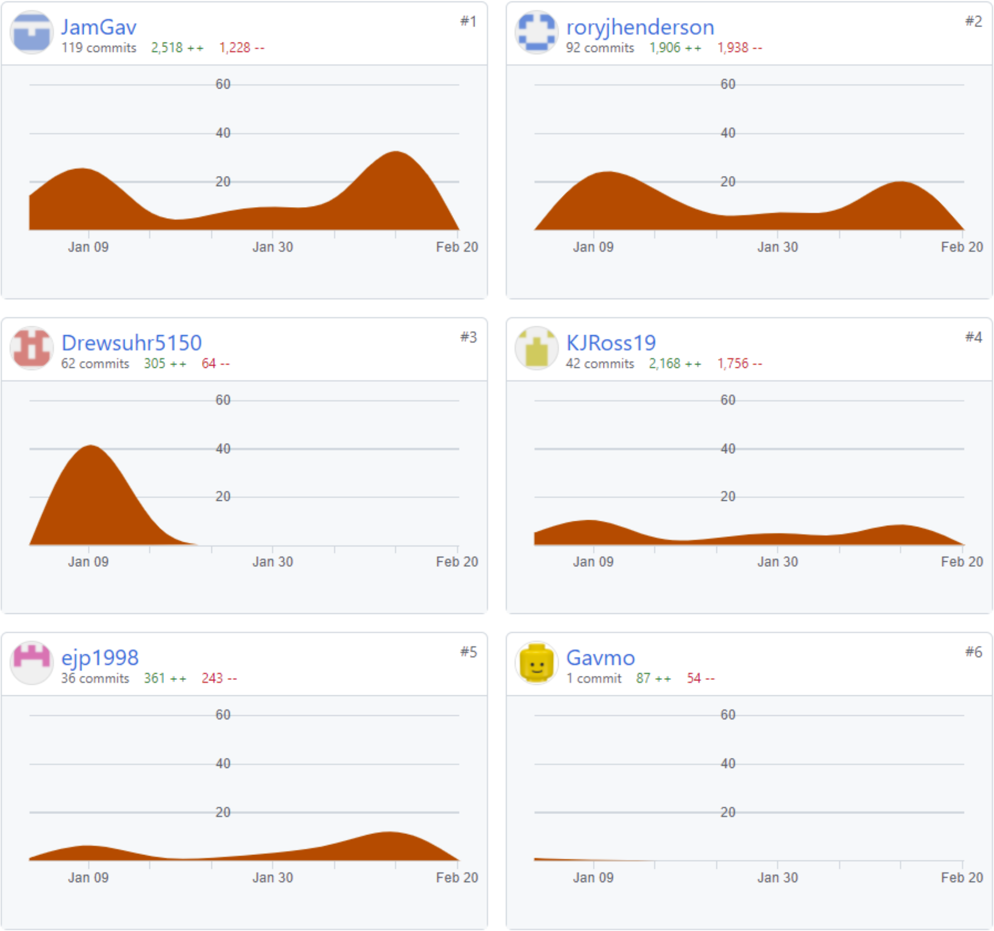
Below is the Commit history of our Github.



You can see many commits at the beginning of the projects as the hub was forked from A2. This was caused by reformatting, group profile updates and other project areas copied or modified from A2.

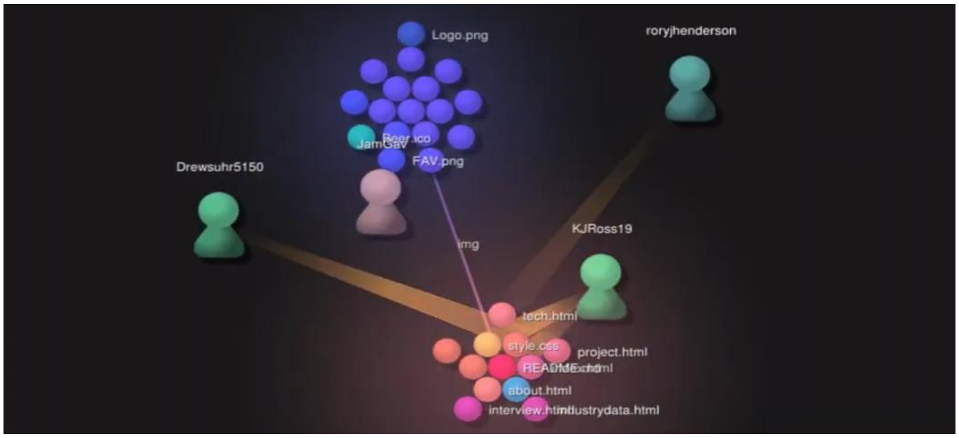
There is then a “quiet patch” as development was happening off GitHub. The team was still in communication and aware of each other’s progress through our Team’s communication.

Towards the end of the project there is another bump in commits as Team members commit their contributions and finalise the site.



You can see in this individual breakdown that the same structure of commits is followed by all active team members.

We also created a Gource visualisation of our commit history for A2 and A3. This illustrates the point above that while it may look like certain members are less active in the project, it does not mean that they have contributed less. The visualisation shows commits from the end of A2 onwards.



This can be viewed at https://www.youtube.com/watch?v=D60OfXThn_g.