

## Context

To navigate life in Singapore, residents should be able to easily find information regarding government policies and schemes.

## User problems

Currently, residents may have difficulty in locating and understanding the information they want due to the following issues:

Problem	Size	Remarks
They don't know the government agency or public service for their issue.	S	They can leverage search engines to identify the correct agency. This initial search may cause frustration and may not be accurate.
They cannot find the correct information on the government website. If it's on the wrong website, the frustration will build in their search to the next website.	M	Without enough context, it can be hard to search the website. Government websites are also not purely informational and may contain news or marketing that is irrelevant to the user's issue. Users must familiarize themselves with each website they attempt to find the information.
They may have to collect information across government websites to handle their issue (e.g. passing of a family member may require information from HDB, CPF, and more).	L	Manually gathering the information to handle one issue is tedious. Information from different agencies may have different contexts and jargon, and possibly create confusion.

## Product vision

Create and maintain<sup>1</sup> a unified consistent self-serve government knowledge base<sup>2</sup> as a single source of truth for residents.

## Potential product

- Articles on a unified website with images and videos (wiki or KB)
- AI personalized newsletter sent with info based on user's data and events
- AI chatbot that simplifies search and can personalize responses
- Agency-agnostic support form
- Platform for live webinars
- Hybrid approach between these

## Selection/prioritization of Minimum Viable Product ("MVP")

To address the user problems raised, the product must be **unified** (i.e. the information must be consolidated from the different agencies), **consistent** (i.e. the information must be written/displayed in a similar fashion and organized in a similar way), and **easy to navigate** (i.e. the user must find the interface easy and intuitive to use).

I assume the average resident is most familiar with a website and that a website is the simplest to implement as an MVP. The consolidated and refined content and the user data collected from the MVP can still be the basis for other solutions in future. For example, the chatbot can use the MVP refined information, and the agency-agnostic form can base its fields using the user data on the information architecture. Hence, I've decided to start with **articles on a website** as a product for this assignment.

## Product plan

Phase 1: Initial MVP and test

<b>Components to build</b>	<ul style="list-style-type: none"><li><b>Front end:</b> the website that users will see.</li><li><b>Analytics:</b> gathers feedback and tracks user behavior.</li></ul>
<b>Content work to do</b>	<ul style="list-style-type: none"><li><b>Style guide:</b> makes all content on the website consistent. This can include jargon used.</li><li><b>Information architecture<sup>3</sup>:</b> the categorization of content, including tags. This should be user-first.</li><li><b>Scope:</b> ensures articles aim at clarity and does not include other types of information (e.g. marketing).</li></ul>
<b>Steps to take</b>	<ul style="list-style-type: none"><li>Gather initial test content from agreeable agencies and adjust the content based on style guide.</li><li>Test front-end design and the content with users.</li></ul>
<b>Success metrics</b>	<ul style="list-style-type: none"><li><b>Helpfulness score:</b> the ratio between <b>Yes</b> and <b>No</b> in the feedback forms.</li><li><b>Traffic:</b> sessions, unique visitors, page views, referral sources, search query, time on page.</li><li><b>Search metrics:</b> average number of searches, searches with no results.</li></ul>

Phase 2: Expansion of MVP and enable agencies with tools to onboard

<b>New components to build</b>	<ul style="list-style-type: none"><li><b>Content management system:</b> allow writers to create, edit, and publish articles.</li></ul>
<b>Content work to do</b>	<ul style="list-style-type: none"><li><b>Migration:</b> migrate and update content. This includes updating information architecture.</li></ul>

<sup>1</sup> Changes must be made in a timely manner. To reduce operations, consider using AI to help identify and update articles impacted by government announcements.

<sup>2</sup> Through continuous testing, pivots, and iterations, the knowledge base need not be a wiki or a website, but possibly grow to be a self-serve chatbot or another solution.

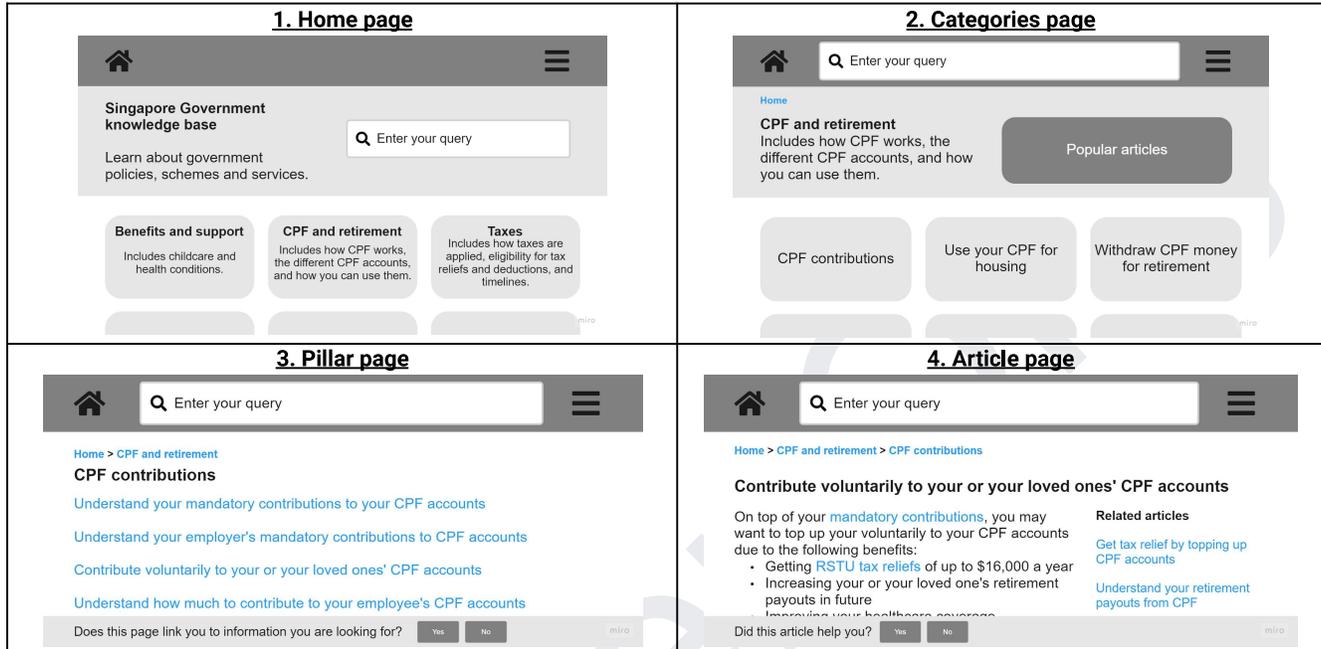
<sup>3</sup> This requires user testing with a card sorting component later on. My initial brainstormed categories are: Benefits and healthcare support, CPF and retirement, Parenting and childcare, Crime and the law, Education, Business and self-employment, Working as an employee, Housing and community, Money and taxes, National Service, Passports and visas, and Citizenship and PR.

<b>Next steps to take</b>	<ul style="list-style-type: none"> <li>Get buy-in from government agencies using metrics gathered from MVP.</li> </ul>
<b>New success metrics</b>	<ul style="list-style-type: none"> <li><b>Migration rate:</b> number of articles migrated from existing agencies to the new website.</li> <li><b>Publish/update activity:</b> number of new articles published and existing articles updated.</li> <li><b>Forms submitted or requests made:</b> users who took action on their issues via the website.</li> </ul>

## Features and design

For the purposes of this assignment, I will focus on Phase 1's MVP that consists of two main components: **front-end** and **analytics**.

Front-end page mock ups



Prioritized features

Feature	Why prioritize	High-level requirements
<b>Search</b> Front-end and Analytics	Basic feature of most websites  Help users find articles related to their query	When the user clicks the <u>search field</u> and enters a <b>search term</b> : <ul style="list-style-type: none"> <li>A dropdown menu will appear and populate with likely article names.</li> <li>This should be a fuzzy search.</li> <li>The user clicks a <b>result</b> to go to the article directly.</li> <li>If they tap <b>Enter</b>, it leads to a pillar page with the search results.</li> </ul> Tracks search metrics in the <a href="#">Success metrics section</a> above.
<b>Breadcrumb navigation</b> Front end	Help users to move up and down the categories	The <u>breadcrumb navigation</u> shows the user the layers up to the current page to give them context on the information. They can click a <b>layer/category</b> to see the content.
<b>Related articles</b> Front end	Proactively suggests articles that can answer the user's query	On article pages, a <u>Related articles</u> section will display links to other articles. To start, this can link to articles in the same category. In future iterations, editors can manually select articles to show based on user feedback and later on, use machine learning to curate likely articles the user is looking for.
<b>Feedback form</b> Front end and Analytics	Data to refine the product and determine its success	When the user scrolls through a pillar page or article, they can see a <u>feedback form</u> at the bottom of the page with <b>Yes</b> and <b>No</b> buttons: <ul style="list-style-type: none"> <li><b>Pillar page:</b> "Does this page show you the article you are looking for?"</li> <li><b>Article:</b> "Did this article help you?"</li> <li>When the user clicks the <b>No</b> button, a dialog box will appear for users to share what they expected to see and to share any further feedback in a free text field.</li> </ul>
<b>Tracking</b> Analytics	Data to refine the product and determine its success	The tracking code and cookie for the website must track the traffic metrics in the <a href="#">Success metrics section</a> above.

Feature backlog (Phase 2)

Front end features		Content management system		
Glossary page	Version history	Rich text tools	Comments	Analytics tools
FAQ page	Table of contents	Save and publish	Versioning	Translation