As HOTB Software Solutions

Experience Pillars - Knowledge Management Tool

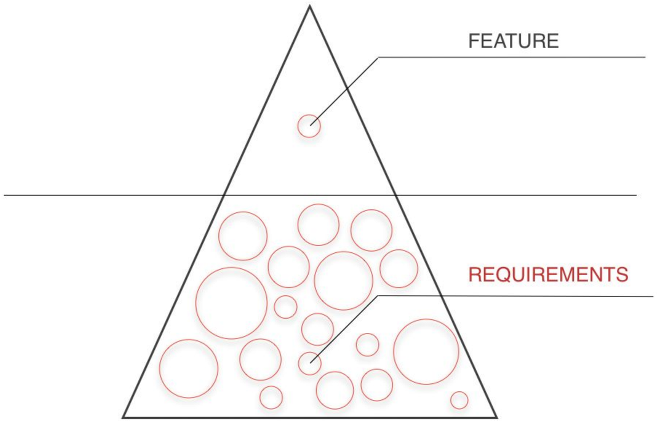
Overview

Your users' worlds are inevitably more complicated than what is observable on the surface. This exercise is a framework to help you strive to understand what you're asking your users to do and the impact it will have on them.

Key User Experiences

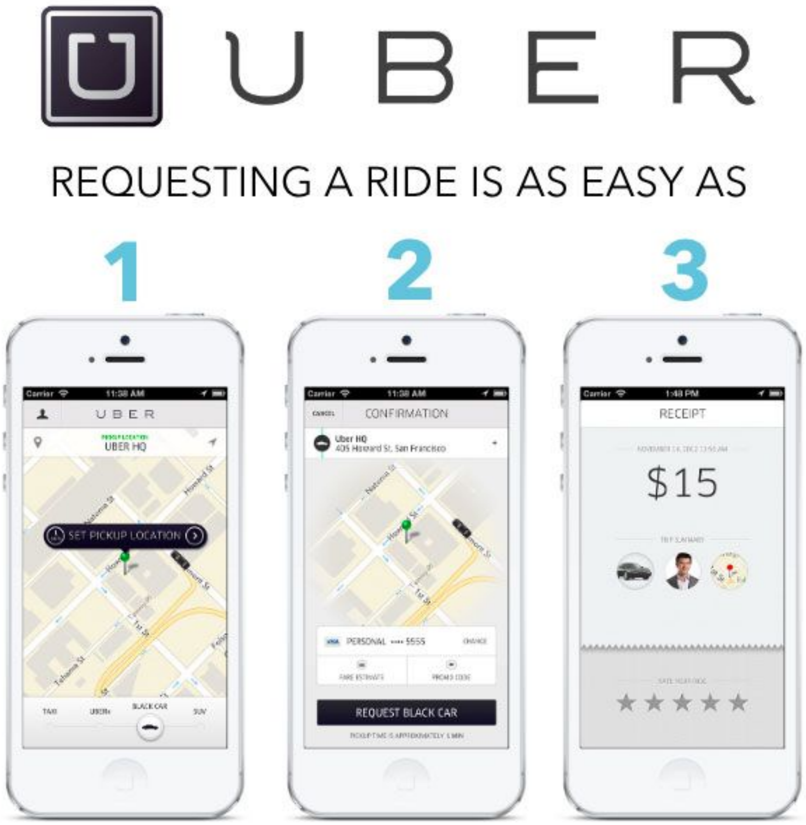
Software and service are always framed by user experiences. These key experiences are foundational pillars that serve as the cornerstones of your platform. Within each pillar lie bricks (features) that when stacked together complete your platform while solving an unmet need.

Properly executed user experiences will layout the golden path your users will follow from login to execution of some intended result. This path should be simple, intentional, and incite an emotional connection resulting in a customer that not only wants your service but requires it on a continual basis.



Software development and feature building is much like an iceberg, with only a small portion exposed with the larger portion remaining hidden from view. The simple 3 step example below will have over a dozen features and potentially thousands of engineering requirements to make it work seamlessly. (Don’t worry that is our job to understand what those engineering requirements are!) What we need from you is the idea, logic, and the intended user behavior as they engage with your platform.

EXAMPLE

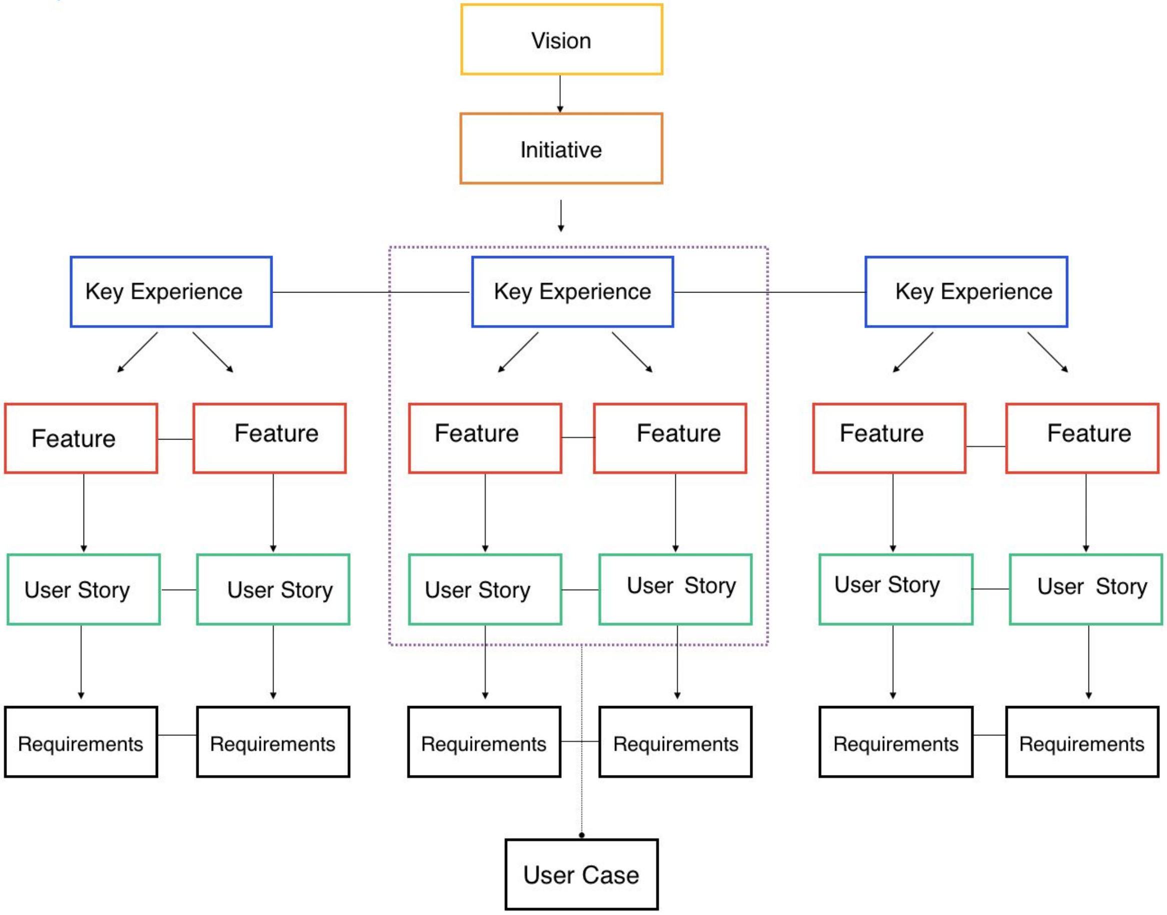


Let's look at an example that may be familiar, UBER.

- Vision: Transportation as reliable as running water, everywhere for everyone.
- The unmet need: Getting a ride to a location when you don't have a car.
- Key Experience: Connecting a rider and driver in real time for an on-demand ride
- Feature
  - User Story: As a person without a car, I want to be able to instantly get a ride to a desired location, so I don't have to get my car, drive, and park.
    - Feature Description: Ability to locate my position in real time and send request to driver with my location.
  - Business Requirement: Rider must input credit card for seamless payment
  - Functional Requirement: Instantly find my gps coordinates

    - Non-Functional Requirement: GPS coordinates must be with 2’ radius and attributed to an address
    - Technical Requirement: Closest driver to be notified within 200 milliseconds of request
    - Acceptance Criteria: Driver showed up at location within the time limit - Pass / Fail
- Dependencies: Stripe for payment is dependent on Apple Pay

Idea Documentation and Structure



Below we have provided an exercise to extract your idea from your mind into a working document. The framework is designed to look at individual features and how they combine and impact many aspects of your platform. Please do the following:

Product Overview:

Build a working prototype that will allow State Agencies/Departments to capture, archive, and manage organization knowledge from its employees and programs. To accomplish that effectively, the working prototype should provide 3 experience pillars centered around; knowledge creation, knowledge sharing, and knowledge development.

For Knowledge creation, users should have the ability to easily create ‘knowledge articles” (KA’s). These can be original records (e.g., specific work instructions or content) and/or packages of content, including documents, user configurable forms, tables, and workflows). In addition, have the ability to provide multiple levels and formats of information in KAs (e.g., bullet points for senior technical levels, scripted specific details for junior/non-technical staff). Allow for role-based security access, to allow control of access and level of information by login.

For knowledge sharing, the working prototype should: Allow for the promotion of process and information across systems and channels, as required, have the ability to create user-defined rules for creation (e.g.,mandatory fields) and lifecycle management (e.g., who, how, when revised and updated) and finally, trigger escalation processes (e.g., automated emails/texts to approvers, reminders) for lifecycle activities.

For knowledge development, the working prototype should: Have the ability to update and improve KAs and access the value of usage as input to predicting new records or record types. Lastly the application must show innovation by learning from existing records (e.g., types, content, usage) and prompting to create new KAs.

Vision Statement:

1. Build a robust and centralized Knowledge Management Tool (KMT), that will allow all State Agencies/Departments to develop, capture, achieve, and manage organizational knowledge from it's agencies and employees.

Goals:

Platform Goals: (hypothetical goals)

1. 35% Increase in the number of published articles across all agencies
2. 50% increase in social media sharing by the public
3. 25% interagency sharing of articles
4. 75% increase in the number of people contributing to article creation

HOTB Goals:

1. Build knowledge management tool in 4 weeks

Key Experience Pillar # 1:

1. Key User Experience:
    a. Build a Knowledge Management Tool that promotes Knowledge Creation.
2. User Stories + Features:
    i. User story 1: As a government employee I want to be able to create articles so I can publish them for internal government purposes and/or publicize them for the general population.
        1. Feature Description: Have the ability to create “knowledge articles”

    ii. User Story 2: As a government employee who is creating articles, I want to have an in-line editor so I can properly format the article while i'm creating my content.
        1. Feature Description: Have the ability to provide multiple levels of and formats in KA’s. (bullet points for senior technical levels, scripted specific details for junior/non-technical staff)

    iii. User Story 3: As a admin or staff member, the platform provides a certain level of inherited access based on my role.
        1. Feature Description: Allow for role-based access, to allow control of access and level of information by login.

    iv. User Story 4: As an article creator, I want to have the ability to create articles with common attributes widely used and understood on the open web.
        1. Feature Description: Article creation elements: Title, Audience, Short Description, Long Description with inliner editor, Tags, and File Uploads.

3. Requirements:
    a. Business Requirements:
        i. The underlying purpose of the platform is to easily create articles for distribution.

ii. Provide a wysiwyg editor so government employee don’t have to use microsoft word or google docs to format articles.

iii. Platform must have different levels of access.

iv. Articles must follow formats generally scene on the web.

b. Functional / Non-Functional Requirements:

i. Articles can only be created by staff.

ii. Articles can only be created for your agency of record.

iii. Articles can only be viewed by the public if they are designated by the author as “public”.

iv. Wysiwyg editor is only used for long description.

v. Attachments can be downloaded by staff, admin, and public (if it’s a public article).

vi. Only a staff member that has been granted access can upload attachments.

vii. Only one admin on the prototype platform - backlog will have multiple admins.

c. Performance Requirements:

d. Technical Requirements:

e. Acceptance Criteria:

i. Only staff can create article. All areas of creating an article for Admin should be greyed out.

ii. When creating an article only the agency associated with the user is available to create articles.

iii. Make sure all elements of the wysiwyg editor are working for long description (only).

iv. Download an attachment

v. Only one admin and create multiple staff.

f. Dependencies:

Key Experiencer Pillar # 2:

1. Key User Experience:
    a. Build a Knowledge Management Tool that promotes Knowledge Sharing.
2. User Stories + Features:
    i. User story 1: As a user of the KA tool, I want to see the articles I have created and have the capability of viewing other articles created on the platform and when necessary take action.
        1. Feature Description: Allow for the promotion of process and information across systems and channels, as required.

    ii. User Story 2: As a user with a defined roll in the system, the platform should adapt by changing certain rules and workflows based on my access level.
        1. Feature Description: Have the ability to create user-defined rules for creation and lifecycle management.

    iii. User Story 3: As a user of the platform with admin access, I want to automatically disposition articles statuses based on my actions so the system can provide a logical workflow for all users to understand the state of the article.
        1. Feature Description: Trigger escalation processes for life cycle activities

    iv. User Story 4: As a staff member, admin, or pubic, I want to be able to share the article via FB, Twitter, and Email when necessary, so I can share what I am reading and learning with my friends and family.
        1. Feature Description: Allow for 3 different types of sharing modalities: Facebook, twitter, and email.

    v. User Story 5: As as admin I want to be able to view, approve or decline articles, so I can quickly understand where I am at with the approval process.
        1. Feature Description: When creating articles, they are automatically routed and dispositioned according to user actions and roles.

When staff creates an article it is routed to a manager to review, approve, or delete.

vi. User Story 6: As a government staff member I want to have a dashboard loaded with pertinent information related to the actions I have taken on the platform so I can make logical and informed decisions about sharing, creating, and developing knowledge.

1. Feature Description: Staff and Administrators will each have a dashboard with the function to view important indicators (KPI’s, metrics), see relevant article information, check status, and take action when necessary.

    a. Staff:

        i. My Article Metrics
        ii. Trending Articles
        iii. My published articles
        iv. My non-published articles

    b. Admin:

        i. System Metrics
        ii. Articles To Review
        iii. Approved Articles
        iv. Declined Articles

3. Requirements:

a. Business Requirements:

i. Support multiple modalities of sharing.

ii. System actively promotes and makes it easy to share across different platforms.

iii. Platform to support public consumption of articles created by different government agencies.

iv. The platform must have workflow rules and processes.

b. Functional / Non- Functional Requirements:

i. Support FB, Twitter, and Email

1. Sharing on Twitter and FB are only allowed for articles that are dispositioned as public and have been publicized.

ii. Both admin and staff can view articles from their dashboard

iii. Only admin can change a status of an article

iv. Only a staff member can edit an article they have written.

v. Only admin will receive notifications

vi. Only an admin can edit/create/delete/add another user. Admin can only add staff not another admin.

vii. Admin has three workflows he/she is responsible for:

1. Articles to Review
    - a. These are articles that are pending and need action
        - i. Status: Pending
2. Declined Articles
    - a. These articles were declined for a specific reason and will not be published.
3. Approved Articles:
    - a. Articles that are now tracking shares and views and can be viewed by selecting article.

viii. Staff's Dashboard contains:

1. Trending Articles - These are global trending articles across the entire platform.
2. My published articles
    - a. Articles the staff member has authored.
3. My unpublished articles
    - a. Articles that were declined
        - i. Statuses: Declined

ix. KPI's:

1. Staff: (My Article Metrics) - Centralized metrics only relevant to the staff's articles.
   a. My Articles Published
   b. Articles in Review
   c. Articles Declined
   d. Views - Calculated as the number of views of article by the public, staff, or admin views
   e. Shares - Total number of shares across all forms of sharing
2. Admin: (System Metrics) - Global view of platform metrics
   a. Articles published - Total number of articles published on the platform.
   b. Articles in Review - Total number of articles currently in review on the platform.
   c. Articled Declined - Total number of articled that have been declined on the platform.
   d. Shares - Total number of shares by all modalities from both internal platform users and the general public.
   e. Total Users - Total number of users on the platform.

c. Performance Requirements:

d. Technical Requirements*:*

e. Acceptance Criteria:

i. Share on FB and view Post.
ii. Share on Twitter and view Tweet.
iii. Share view email and receive email.
iv. Both admin and staff can click and view articles.
v. Admin receives email notification when a new article is created.
vi. Admin creates user and they receive an email to create account.
vii. Admin deletes user and the user no longer has access or is visible in the manage users table.
viii. Admin to edit users information and its changed and saved.
ix. Saff article workflow:

1. Staff creates new article then it moves to “article to review” in admin dashboard.
2. Admin declines article and it moves to “my non-published” article on staff dashboard.
3. Admin approves article and it moves to “my published articles” in staff dashboard.

x. Admin Article Workflow:

1. Declines article and it moves to declined table on dashboard.
2. Admin approves article and it moves to approved article table on dashboard.

xi. KPI’s

1. Staff:
    a. Calculate in accordance to the write up above, under “functional requirements”.
2. Admin:
    a. Calculate in accordance to the write up above, under “functional requirements”.

f. Dependencies:

i. Facebook
ii. Twitter
iii. Email service provider

Key Experiencer Pillar # 3:

1. Key User Experience:
    a. Build a Knowledge Management Tool that promotes Knowledge Development.
2. User Stories + Features:
    i. User story 1: As a user of the KA, tool I want the system to learn from my behavior and proactively make suggestions so the system is easier to use and more robust overtime.
        1. Feature Description: Have the ability to update and improve KA's and access the value of usage as input to predicting new records or record types.

    ii. User Story 2: As a creator of the KA tool, the system should globally learn from all article records so the system can be improved over time.
        1. Feature Description: Show innovation by learning from existing records

    iii. User Story 3: As a government employee I want to see trending articles based on shares and views so I easily create similar articles that may make the same impact internally and with the pubic.
        1. Feature Description: Articles that have algorithmically (shared,viewed) started to trend will be displayed on the dashboard and promote additional article creation by providing a button to create similar.

    iv. User Story 4: As a creator of an article, I want the system to suggest previously used tags provided by other users, so I may create and share articles that are easy to find and are of value.
        1. Feature Description: When creating articles the system will algorithmically provide suggested tags based on previous articles that have been written.

    v. User Story 5: As a civilian or government employee I want to easily search for articles in many different ways, so I can quickly locate what I am looking for.

1. Feature Description: The general population and government employees will have the ability to easily search any articles that were dispositioned as “Public”. Article can be searched by Agency, Tags, and title. (Admin can see all article tagged as internal or pubic)

vi. User Story 6: As a staff member or administrator, I want to be able to comment about articles that are pending so they can be approved and published as well as see progressive article history.

1. Feature Description: When reviewing an article, the article author and admin will have the ability to provide feedback by way of adding comments.
2. Feature Description: Each article will show the history:

a. Created on
b. Last Updated
c. Edit By
d. Status change

vii. User Story 7: As a administrator I want the ability to Review, approve, and decline certain articles, so the platform contains only thoroughly vetted and reviewed content.

1. Feature: Articles move from pending, approved, and/or declined

3. Requirements:

a. Business Requirements:

i. The system learns from user behavior
The system makes relevant suggestions when creating new articles
ii. Creating an article should be quick and easy
iii. Feedback on pending articles should be bi-directional between admin and staff
iv. Searching for articles should provide the person multiple ways to find relevant articles

b. Functional / Non- Functional Requirements:

i. System provides better workflow overtime by learning from platform users.

ii. System monitors tags from previously published articles and suggests tags for new articles being written for publishing.

iii. Trending articles:

1. Algorithm for Trending : Trending Score = Views * 1 + Shares*3
2. Once a article is identified as “trending” the user can easily create a new one similar to that one.

   a. When selecting create new, the article will carry over all the same attributes over from the trending article including: Tags, Audience, agency, and sharing.

3. The author is the only person who has the ability to edit his/her own work.
4. Comments are bi-directional between Staff and Admin

iv. Disposition matrix and rules:

1. Only Admin can review, approve, decline an article.
2. Once an admin opens an article it automatically moves to “in review”.
3. The admin will then have the option to comment, approve, or decline.
4. Only a staff member can delete an article that is not yet approved or declined.

c. Performance Requirements:

d. *Technical Requirements:*

e. Acceptance Criteria:

i. Disposition matrix follows: new, in review, approved, declined.

ii. Only author can edit articles up to the moment its approved or declined.

iii. When creating a similar trending article, tags, audience, article type and sharing modalities are carried over.

iv. Trending article algorithm is calculating correctly.

v. Tags that are on previous articles are now showing on the “create article: screen to use on new articles.

vi. Articles are showing the correct article history elements:

1. Created on

2. Last Updated
3. Edit By
4. Status change

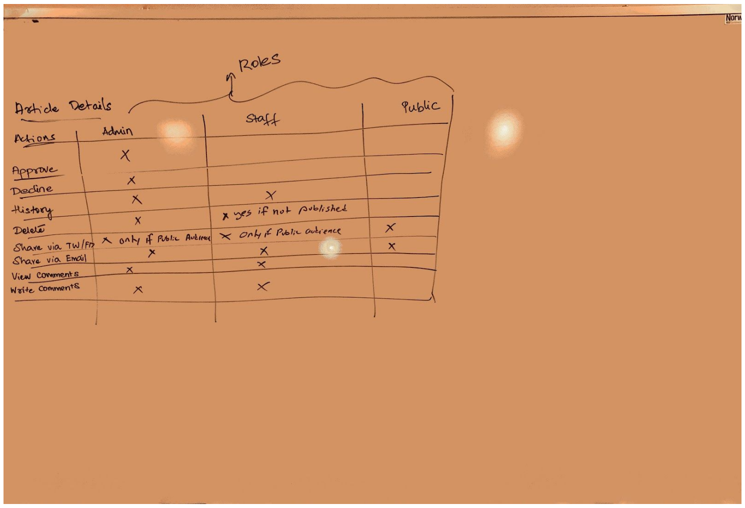
vii. Article comments are showing up on the article page for both admin and staff.

viii. A civilian or gov. Employee can search by tags, agency, or title.

ix. Staff can create trending articles in the trending article section

1. Trending articles will carry over to new article:
   a. Tags, audience, Share, article

x. Article Detail Roles:

1. 

Roles

Article Details Actions	Admin	Staff	Public
Approve	X		
Decline	X		
History	X	X	
Delete	X	X yes if not published	
Share via TW/FB	X only if Public Audience	X Only if Public audience	X
Share via Email	X	X	X
View Comments	X	X	
Write Comments	X	X	

f. Dependencies:

Key Experiencer Pillar # 4:

1. Key User Experience:
   a. Brief Description: General System Setup
2. User Stories + Features:
   i. User story 1: As a platform user I want to be able to easily create a login, ability to view/edit my profile and if given the authority be able to change system settings.
      1. Feature Description: User creation through email and password.
      2. Feature Description: User will have a profile section that will provide them the ability to edit:
         a. Name information
         b. Email
         c. PW reset
         d. Notifications: (only on the admin profile)
            i. Turn on off notifications for email
         e. Upload Attachments
            i. Can be turned on/off by admin when creating users
      3. Feature Description: Administrators will have access to system wide settings that give them the ability to edit/change:
         a. Manage users: Create/edit/delete users
         b. Article Settings: Ability to turn on/off the two types of article templates: Public, Private (Backlogged)
         c. Notifications: Ability to globally turn on/off: sms, and email for the platform. (Backlogged)
3. Requirements:
   a. Business Requirements:
      i. *Setup should be easy and use standard protocol seen throughout the web*
   b. Functional/Non-Functional Requirements:
      i. Only admin has access to settings
      ii. Only the user can request PW reset
      iii. Notifications:

1. Only Email
2. SMS (backlogged)
3. Admin can only turn on and off email notifications
4. Notifications are triggered when a new article is created.
    a. Notification is triggered when an article is moved from New to Approved or Denied. (Backlogged)
5. Comments trigger an email that goes to the other person. The user would then click the link in the email and deep link to that comment. (Backlogged)

iv. Only admin can create a user

v. Article settings allow only the admin to toggle on/off public or private templates. Each template provides a different set of viewable fields. (Backlogged)

1. Public:
    a. No History
    b. No Comments
2. Private:
    a. History
    b. Comments

c. Performance Requirements:

d. *Technical Requirements:*

e. Acceptance Criteria:

i. Create a user via email.

ii. Ability to request a change of PW.

iii. Ability to change password.

iv. Notifications that are enable/disabled are either activated or no longer triggered

v. Admin can create a new user

vi. Admin can edit user information

vii. Admin can delete a user

viii. Public articles are limiting the view by removing history and comments

ix. Admin creates user that has the ability to upload attachments removed - no upload capabilities on create article

x. Admin creates user that has the ability to upload attachments - upload capabilities are restored on create article.

f. Dependencies:

i. Amazon SNS

ii. Email service