Trinity WayFinders

WayFinder Application

Requirement Specifications Document

Revision History

Version	Created Date	Author	Comments
1.0	7-2-2019	Nicky Bonello	

Authors List

Sl. No.	Name	Role
1.	Nicky Bonello	
2.	Saad Tariq Malik	

Table of Contents

Table of Contents

1. Objective

This document contains the details of all the functional behaviour of the Trinity Wayfinder application.

2. Requirements Overview

The following sections will contain all the use cases for the two categories of users (i.e. the Admin User and the End-User).

3. Requirements Classification Methodology

The Requirement IDs were allocated to tasks automatically by JIRA* based on the order in which they were added.

The following priority levels have been defined for the requirements:

1) Must-Do: To identify requirements integral to our application
2) Highest: To identify requirements that are of extremely high utility to the user but not critical to the overall operation of the application
3) High: To identify requirements that yield high utility to the user (but lower compared to the 'Highest' requirements)
4) Medium: To identify requirements that could serve as nice add-ons/extra features that a mature application would likely have.
5) Low: To identify nice-to-have features that would improve the application
6) Lowest: To identify nice-to-have features that we might consider in the future but currently not in the scope of the project.

* JIRA is an online tool for bug tracking and agile project management.

4. Functional Requirements
4.1. User Requirements

Table-1

Req-ID	Requirement	Priority
WR-53	A user should be able to cancel their journey whenever they desire.	Must Do
WR-50	A user should be given multiple options for their desired journey	Must Do
WR-68	A user should be able to use the application even when there is an intermittent interruption in the network (internet).	Must Do
WR-64	A user should be able to specify their journey as from A to B (not from their current location).	Must Do
WR-62	A user should be able to specify their journey as from their current location.	Must Do
WR-59	For each route, the options presented to the traveller should include three or more transport modes.	Must Do
WR-54	A user should be able to modify their journey en-route if they desire.	Must Do
WR-57	A user should be given information about the emissions of their chosen route and if applicable, given an alternative option with lower emissions.	Highest
WR-56	A user should be given an update to their rout	Highest
WR-60	The user profile should be updated with incentives based on the choice of routes.	Medium
WR-66	A user should be able to specify an arrival time and be given the most sustainable journey based on this time.	Low
WR-65	A user should be able to specify a multi-hop journey.	Low
WR-52	A user should be given Dublin Bikes a suggested mode of transport if they have said that they have a Dublinbikes account.	Low
WR-86	A user should be able to disable their account	Lowest
WR-85	A user should be able to specify how many people they are travelling with	Lowest

WR-67	A user should be able to send a report requesting help.	Lowest
WR-73	The sales team should have scenarios of application usage using simulated data for demonstration	Must Do
WR-72	The sales team should have a feature using which they could simulate a negative event during the route and show the functionality of dynamic rerouting	Must Do
WR-70	Provision should be available to stream live pollution data.	Must Do
WR-81	A user should be able to delete account if desired.	Must Do
WR-80	A user should be able to request password if forgotten.	Must Do
WR-79	A user should be able to change their password.	Must Do
WR-78	A user should be able to login and manage their profile.	Must Do
WR-77	A user should be able to sign-up for an account	Must Do
WR-82	During sign-up the user should be asked to agree/not to agree to the following:	Must Do
WR-2	A user should be able to sign-up for an account using the following	High
WR-84	A user should be able to sign-up and sign-in using Google and Facebook Accounts	Low
WR-3	A user should be able to specify their top three requirements for travel	Must Do
WR-22	A user should be able to turn on/off location.	Must Do
WR-19	A user should be able to control their notification settings	High
WR-87	A user should specify whether they are employed/student or other at sign up	Medium
WR-24	The user should be able to view their contributions and history	Medium
WR-18	A use should be able to bookmark / save favourite routes and access it.	Medium
WR-23	A user should specify whether they have a Dublinbikes account.	Low
WR-20	A user should be able activate/disable active re-routing	Low
WR-25	A user should be able to delete part or all of their contributions or history.	Lowest

WR-21	A user should have the choice to display / disable the virtual incentives publicly.	Lowest
WR-13	A user should be able to add an 300 character profile description	Lowest
WR-10	A user should be able to put in a profile picture	Lowest

4.2. Administrator Requirements

Table-2

Issue key	Requirement	Priority
WR-15	System configuration - backend environment configuration editing.	High
WR-12	Service APIs Management	Must Do
WR-11	Service Coverage Management - Manage the coverage of the service and constraint userâ€™s assessment.	Low
WR-14	Incentives management - e.g., assign badges to user based, active discount coupons, collecting points and exchange points.	Lowest
WR-9	Query - Query a admin User	Low
WR-8	Modify - Change Admin User Profiles	Low
WR-7	Delete - super admin can delete admin user	Low
WR-6	Add Admin User- Super admin can create a new account for an admin	Low
WR-4	Query - Query user	Highest
WR-5	Inactive - inactivate a userâ€™s account.	High

5. Non-functional Requirements

Table-3

WR-45	The system should be available 24/7	Must Do
WR-44	Pre-loaded data should be used in absence of network	Must Do

	connectivity.	
WR-43	System should automatically reconnect as soon as network connectivity is available.	Must Do
WR-47	System should be compatible with the following operating systems	Low
WR-46	System should be available for all major web-browsers	Low
WR-58	All the Dependencies/Requirements should be documented well	Must Do
WR-55	The system architecture document should have detailed architectural details	Must Do
WR-49	All the user workflows should be documented well	Highest
WR-63	The license should be documented and available to all users	Low
WR-51	An effective troubleshooting guide should be maintained	Low
WR-48	There should be clear Setup instructions for the application	Low
WR-61	The documentation should be automated whenever possible	Lowest
WR-41	Seamless operation during poor/no network connectivity	Must Do
WR-40	Real-time route suggestions from our system.	Must Do
WR-39	The system must handle x (large number) of users.	Must Do
WR-71	The system should be tolerant to Failure: crashes, incorrect routes,	Highest
WR-69	There should be a measurable way to minimize MTBF (mean time between failures)	Medium
WR-83	The application should be able to scale based on the load	Must Do
WR-37	Passwords shall never be viewable at any point.	Must Do
WR-35	The guest user should have access to frontend without signup	Must Do
WR-34	No personal data should be collected from the Guest User	Must Do
WR-33	The user shouldn’t be able to access/modify other users data	Must Do
WR-32	The user can only delete personal data	Must Do
WR-31	The user can use backend services but shouldnâ€™t be able to modify	Must Do

WR-30	The user has access to personal data through frontend.	Must Do
WR-29	The user should not have API and database access directly	Must Do
WR-28	The admin should have no access to userâ€™s personal data	Must Do
WR-36	All network communication must be done over a secure channel.	Highest
WR-38	Users should be able to change their passwords.	Medium
WR-27	The superadmin should be able have Read/write access to API, backend services, frontend.	Low
WR-74	The system should have a easy to user interface	Highest
WR-75	The APIs should have inbuilt documentation for easy usage	High
WR-88	Multilingual Support	Lowest
WR-76	The User Interface should be adaptable to users reading needs	Lowest

6. Assumptions, Constraints and Dependencies

Constraints:

1) Time: The project's duration is 12 weeks (1440 man hours).
2) The application will be developed using XP Principles.

Assumptions:

1) The team would be able to accommodate any changes in the requirements
2) People would be mainly using the application for long distance commutes

Dependencies:

1) Required 3rd party tools are available
2) Development infrastructure is available

7. References

Brandenburg, L. (2019). *What Goes Into a Functional Specification?*. [online] Bridging-the-gap.com. Available at: https://www.bridging-the-gap.com/functional-specification/ [Accessed 7 Feb. 2019].

Calpolyedu. 2019. Calpolyedu. [Online]. [7 February 2019]. Available from: http://users.csc.calpoly.edu/~jdalbey/SWE/QA/nonfunctional.html

Altexsoft.com. (2019). *Functional and Non-functional Requirements: Specification and Types*. [online] Available at: https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/ [Accessed 7 Feb. 2019].

Bcsorg. 2019. Bcsorg. [Online]. [7 February 2019]. Available from: https://www.bcs.org/upload/pdf/non-functional-requirements-091214.pdf

8. Appendix

Jira Dashboard - http://35.246.117.35:8080/secure/RapidBoard.jspa