Abstract

I am doing the Offline HTML5 Maps Application project because I have a keen interest in software, particularly on mobile platforms, that utilises map technology to provide a user with helpful information, especially contextually relevant information based on their current location. It would be worthwhile to apply this interest using web technologies instead, experimenting with the combination of HTML5 mapping and local storage, while still offering functionality traditionally only offered with online map services, such as providing directions and information about places of interest. The final project should be an offline HTML5 application that is capable of storing and displaying richly detailed vector maps with a wealth of information and functionality a user would expect from a native application.

I believe this would be a useful project that currently faces minimal competition. For example, Google’s Maps mobile application is a native application, hence only usable on certain platforms, and currently only allows downloading and viewing offline certain areas. Furthermore, it does not currently support search or directions functionality offline.

The project reports I write throughout the first term will typically revolve around the sorts of topics I have been researching and how I have used them when developing either proof of concept programs or the final project. These will include key background concepts and technologies relevant to web development and mapping tools.

Timeline

I aim to develop four proof of concept programs before the end of the first term, accounting for half of the early deliverables:

1. “Hello world” offline HTML5 application: A simple application built using HTML5 that is viewable offline.
2. To-do list application using an indexedDB: A simple application to demonstrate the use of an indexedDB for storing data offline.
3. Drawing shapes using the HTML5 canvas: An application that allows the user to draw shapes within a HTML5 canvas.
4. Webpage that loads and lists raw OpenStreetMap data: A demonstration of retrieving and displaying OpenStreetMap data in a raw format.

These will be coupled with relevant reports on topics relating to what was needed to develop each program.

Term 1:

|  |  |  |  |
| --- | --- | --- | --- |
| **Delivery date** | **Development** | **Report** | **Notes** |
| 9th October | “Hello world” offline HTML5 application | Basic webpage development: Short write-up on HTML, CSS, JS | I am familiar with using these languages, so I expect developing a basic offline application to take relatively little time. |
| 16th October | To-do list application using an indexedDB |  | I have not used indexedDBs before and so need a week to understand it and implement it. |
| 23rd October |  | HTML5 offline technologies: Basics of caching and indexedDB | I will write a little on what I have learnt about indexedDBs and other methods of caching data. If this is finished quickly I can move onto HTML5 canvases sooner. |
| 30th October | Drawing shapes using the HTML5 canvas | Drawing shapes: Using the HTML5 canvas to draw shapes in the browser |  |
| 6th November | Webpage that loads and lists raw OSM data |  |  |
| 13th November |  | OSM data representation: Comparing and contrasting vector and image tile maps | After playing with downloading and displaying raw OSM data, I want to spend a week researching how to transform this data into visual maps. |
| 20th November | Load and display map data online and offline, probably limited to a specific area at first like London |  | Using what I have learnt from the early deliverables, I will mainly focus on completing basic versions of the final deliverables for the remainder of the term. I may write short reports on what I have worked on when I reach key milestones, such as completing one of the deliverables. |
| 27th November | Extra information on request |  | Play with supporting basic uses of the user tapping on certain areas of the map and relevant information being displayed, possibly retrieving extra data from other sources. |
| 4th December | Basic search functionality |  | Allow the user to enter a search term and have the focus of the map move to the location best matching the term. Will probably have a very crude UI at first. |
| 11th December | Polish existing functionality i.e. Improve performance, smooth-line experience of downloading data with the user having to do less |  |  |

Halfway through the first term, the early deliverables should be complete. I aim to have completed the final deliverables, at least with basic functionality and maybe not fully optimised in terms of performance, by the end of the first term.

Term 2:

I have a rough idea of some of the more advanced features I want to implement during the second term.

* Advanced search functionality, including listing various possible matches to a search term and allowing for filtering by certain types of locations.
* Maps covering more landmass, assuming only a limited amount of data is used to begin with during term 1.
* Basic directions from one point to another, calculated offline.

Risk Assessment

One possible key risk associated with the project is handling large quantities of data. It may be difficult to handle, for example, loading and displaying potentially gigabytes of offline map data efficiently and with minimal negative impact, such as lag, on the user. This concern is validated by the fact OpenStreetMap’s own wiki states that there are “huge amounts of data”, a snapshot of the current data for the entire planet being “almost 40 GB compressed”. The best way of ensuring this risk is not realised is probably to thoroughly research preferable methods of translating raw data into vectors, using web technologies, that have less of an impact on performance. A further way would be to use only a limited set of data, focusing on a particular region, at least towards the start of the project.

Another possible risk associated with the project, albeit arguably less serious, is that some users may not be able to take advantage of interacting with maps offline as they may not have a version of a browser that supports HTML5 application caching, such as versions of Internet Explorer older than Internet Explorer 10 according to W3Schools. At this point, it would appear that the best way of dealing with the issue is to ensure that all functionality actually works just as well online as offline, and backwards compatibility should be considered when adding new features to the project, so that at least as much as possible is usable in older or less popular browsers.

Bibliography

* Maps for mobile: Download an offline map

<https://support.google.com/gmm/answer/3273567?hl=en-GB>

This page explains the limitations of using Google’s Maps mobile application offline, hence can be used to justify the necessity of this project.

* Foursquare API: Photos from a Venue

<https://developer.foursquare.com/docs/venues/photos>

This is an example of an API that could be used for retrieving and displaying relevant information for points on the map. The Foursquare API’s photos endpoint can be used to get photos of a particular venue.

* OpenStreetMap wiki: Downloading data  
  <http://wiki.openstreetmap.org/wiki/Downloading_data>  
  A short document useful for acquiring a sense of how much data I may have to work with, and potential methods of downloading it.
* W3Schools: HTML5 Application Cache

<http://www.w3schools.com/html/html5_app_cache.asp>

This page explains which versions of which browsers support caching in HTML5, as well as providing good information on how to get started with implementing caching.