CS3821 – BSc Final Year Project

Noe Chen Jiang

2023

OFFLINE HTML5 MAPS APPLICATION

Project Plan

Supervised by: Francisco Ferreira Ruiz
Department of Computer Science, Royal Holloway University of London

1 Abstract

Offline maps applications are essential for use in areas of limited internet connectivity. These applications were made possible thanks to recent developments in HTML5 such as Web Storage, Cache, Service Workers and IndexedDB.

Cache allows the client to store Web objects (such as HTML files) that can be loaded directly from the machine instead of the server, hence allowing offline functionality as well as improved performance [1]. Another critical component of offline functionality are Service Workers, which act as a proxy between the web application and the network or (most importantly) cache].

Web Storage allows browsers to store key/value pairs on the client side like cookies. However. Web Storage allows storing a larger amount of data and can store more complex data structures, making it more suitable for this project compared to cookies given the complexity of OSM data.

On the other hand, depending on the scope of the project, IndexedDB is also worth considering as it can also handle complex data structures as well as even larger data sets and complex queries. Whether to either use Web Storage or IndexedDB or even a combination of both is in consideration.

One way of representing geographical information is with “vector maps”. This is an improvement over a map of individual pixels (a bitmap), as vector maps describe the shape and locations of each map element (such as lakes and roads). Moreover, vector maps reduces the amount of downloaded data needed, providing a more user-friendly experience [8].

Other relevant technologies are HTML5 canvas and jQuery. The HTML5 canvas element uses JavaScript to draw graphics. It is important to note that directly drawing map elements with OSM data using canvas would be very difficult, so the use of other APIs to make such process less time consuming is crucial. jQuery will also be used, as it solves issues caused by browser incompatibilities [6] and can save development time by simplifying common JavaScript tasks.

This project will leverage these technologies to create a vector map application and will be based on Open Street Map (OSM) data retrieved from the web or stored previously in a local database and rendered in real time. This application will make geographic data more accessible in resource-constrained regions. If time allows additional features such as supporting different OSM formats, downloading map data dynamically when a connection is present, dynamic display of different types of data and searching for interesting features will be implemented.

2 Timeline

During Term 1, the plan will focus on exploring these technologies with proof of concepts documenting any findings for the reports. Moreover, I will also work on developing the application out of term time to leave more time to explore suggested extensions, but only if progress allows. In Term 2, there will be further development efforts in finalizing the application as well as implementing possible extensions.

September-January: Proof of concepts, reports and research

Week 1-3 (late September to early October)
- Project plan: define scope, goals and milestones, as well as research.
- Development environment setup and version control (GitLab).

Week 3-4
- Use Service Workers and Cache to create a "Hello World" offline HTML5 application.
- Document and report findings and challenges.

Week 5-6
- Develop "to do list" application with IndexedDB/Web Storage.
- Experiment with IndexedDB/Web Storage to find which technology is more suitable.
- Document and report findings and challenges.

Week 7-8
- Build an application to draw shapes using HTML5 canvas.
- Document and report findings and challenges.

Week 9-10
- Create web page that loads and list raw OSM data.
- Document and report findings and challenges.

Week 11-12
- Write a report on basic web page development in HTML5: HTML, JavaScript, CSS.
- Write a report on advanced technologies: jQuery, HTML5 canvas.
- Write a report on developing an offline HTML5 application: Web Storage, Indexed DB, Service Workers and the Cache API
- Write a report on Open Street Map data representation and vector vs. image tile maps.
- Include insights from research and proof of concept programs.
- Prepare interim report and presentation.

Week 13-15
- Begin development of the offline HTML5 Maps application and build upon proofs of concepts.
- Implementation of basic functionality: offline access, loading and displaying map data, zoom and map navigation.
- Document progress, findings and challenges.

January-May: Development and extensions

Week 16-18

- Continue development and implementation of basic functionality.
- Document progress, findings and challenges.

Week 19-21

- Develop features for improving rendering performance such as database indexing and image caching.
- Do performance tests and optimize accordingly.

Week 22-24

- Finalize application and conduct testing to ensure requirements are met.
- Fix bugs and address usability issues.
- Consider implementing extensions (only and only if progress and time allows): allowing the dynamic display of different types of data (highlight restaurants, parks, etc.), supporting different OSM data formats, and downloading map data dynamically when a connection is present.
- Test and document implementation.

Week 25-27

- Compile project documentation: reports and code.
- Prepare for project submission and demo/presentation.

3 Risk assessment

	Description	Impact	Probablity	Mitigation
Time management	Inadequate time management can cause delays and timeline extensions, as well as compromising the quality of the deliverables.	Medium	High	Continuous monitoring progress against plan and identify potential time management risks ahead of time and develop mitigation strategies.
Troubleshooting	Possible issues may arise when setting up the development environment and from the use of specific libraries and frameworks	Medium	High	Keeping a bug log and seeking online communities for help when facing issues
Offline functionality	It can be challenging to build robust offline functionality with Service Workers, Cache and IndexedDB	High	High	Begin with simple examples (proof of concept programs) and incrementally build complexity and appropriately test at each step of the process to quickly solve issues that may arise
Cross-browser compatibility	Different browsers might interpret HTML5 differently so compatibility issues may arise	Low	Low	Testing across different browsers and by considering the possibility of automating testing

Performance	Performance benchmarks could be difficult to meet, leading to unresponsive user interface and slow loading times	Medium	Medium	Optimization and identifying and addressing performance bottlenecks
OSM data complexity	OSM data can be complex due to the variety of data formats and structures, making data parsing and display more difficult	Medium	Medium	Developing robust data processing routines and achieving a good understanding of OSM data structures
Scope creep	Expanding the scope of the project, especially with suggested extensions, may cause timeline delays and increased complexity	High	Low	Suggested extensions should only be considered as post-project improvements

Acronyms

OSM – Open Street Map
HTML – Hypertext Markup Language

Glossary

Cookies – Small files of information that temporarily store information.

Bibliography

[1] Wang, J. (1999). A survey of web caching schemes for the internet. *ACM SIGCOMM Computer Communication Review*, *29*(5), 36-46.

Web caching is important to the offline functionality of the application, so it is helpful to know how web caching improves the Web.

[2] MDN Web Docs. (n.d.). *Service worker API* . Web APIs | MDN. https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API

Useful to learn how service workers they might be used in the context of the application.

[3] WHATWG. (2023, September). *HTML Living Standard*. https://html.spec.whatwg.org/multipage/

The HTML Living Standard provides comprehensive information about HTML5, and following web standards ensures that the code is well maintained and more future proof.

[4] W3C. (2023, August 8). *Indexed database API 3.0*. https://www.w3.org/TR/IndexedDB/

[5] Kimak, S., & Ellman, J. (2015, December). The role of HTML5 IndexedDB, the past, present and future. In *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)* (pp. 379-383). IEEE.

Learning about IndexedDB will help in deciding whether to use web caching, IndexedDB or a combination of both.

[6] Bibeault, B., De Rosa, A., & Katz, Y. (2015). *jQuery in Action*. Simon and Schuster.

jQuery is crucial in simplifying JavaScript programming and in contributing to development.

[7] Bennett, J. (2010). *OpenStreetMap*. Packt Publishing Ltd.

Understanding OSM will allow me to make an informed decision on the scope of the project, and mitigate scope creep.

[8] Netek, R., Masopust, J., Pavlicek, F., & Pechanec, V. (2020). Performance testing on vector vs. raster map tiles—comparative study on Load Metrics. *ISPRS International Journal of Geo-Information*, *9*(2), 101. https://doi.org/10.3390/ijgi9020101

Information about the performance of vector maps will be useful should optimization be necessary in the future.