

Online Dispute Resolution for Maritime Collisions

Report Name	Outline Project Specification
Author (User Id)	Christopher Benjamin Ashton (cba1)
Supervisor (User Id)	Georgios Gkoutos (geg18)
Module	CS39440
Degree Scheme	G600 (Software Engineering)
Date	February 3, 2015
Revision	1.1
Status	Release

1 Project description

When it comes to resolving disputes, many people are increasingly turning to online dispute resolution (ODS) platforms as an alternative to taking the case to court, settling disputes more quickly, conveniently, and at a lower cost.

These platforms already exist; they allow lawyers to open disputes on behalf of their clients, upload documents and type content in a structured manner, and hopefully reach an amicable resolution. However, there is no business logic that helps influence the outcome of a dispute. Resolution is a manual process performed by the lawyers.

Online Dispute Resolution for Maritime Collisions will attempt to introduce that business logic in an abstract way so that a module containing maritime law business logic can be plugged into the system. It will ask relevant, structured questions, interpret the answers by both parties and play out a "court simulation" indicating the outcome of the case should the dispute be taken to court. It may also retrieve similar historic cases which can be fed into the simulation.

2 Proposed tasks

- **Familiarise myself with ODS** as a concept and where it stands today. How popular is it? Who are my competitors? What functionality is required? Where are ODS platforms headed? [8] [9]
- **Understand the intricacies of maritime law** so that I can validate the behaviour of the application. [2] [1]
- **Develop a simple Online Dispute Resolution Platform** as a basis for further development. This may require building from scratch, or I may be able to reuse open source software. [5] ODS requires a minimum level of functionality, including user registration/login capabilities and the ability to open, add evidence to and resolve a dispute.
- **Extend the ODS platform to take a module of business logic.** The platform should be flexible enough to allow business logic modules to:
 1. **Dynamically render different questions** according to the user's answers to previous questions.
 2. **Perform arbitrary analysis on all of the inputted data**, being mindful of confidentiality issues whereby one party should not be able to see the answers of the other.
 3. **Render results** at the end of the question phase, e.g. court simulation.
- **Define the maritime law business logic module.** This can start off quite small, e.g. by only encoding a handful of rules, before being iteratively developed to take more conditions into account. The underlying ODS platform should be abstract enough to allow any concrete implementation of a module, whether it uses simple procedural processing, a decision tree, or something as complex as a simulated neural network. The development of this module requires investigating the solutions of others [6], spike work and incremental development.
- **Gather historic maritime law cases** in a consistent format, to enable the "retrieve similar cases" functionality. Existing collections are stored in plain text formats [3] [4] [7], so would require manual or natural language processing before storing in a consistent data structure for later reuse.

3 Project deliverables

The main project deliverable, i.e. working software, can be developed iteratively in five stages. The first three refer to developing an increasingly sophisticated ODS platform. The final two stages (to be completed if I have time) refer to developing an increasingly sophisticated *implementation* of the maritime law business logic module:

1. **Build an Online Dispute Resolution platform.** This is the minimum viable product and is potentially substantial enough to be a Major Project on its own, encompassing front-end development, server-side logic and database integration.
2. **Tailor the ODS platform to Maritime Collisions,** providing some sort of conclusion given the details of a maritime collision dispute.
3. **Make the ODS platform abstract, able to take a module of business logic** (perhaps Maritime Collisions, perhaps something else).
4. As an additional feature, the system should be able to **retrieve the most similar historical cases**, which should be a useful reference for the lawyers.
5. Following on from the ability to retrieve similar cases is the ability to **feed the details of the similar cases into the current dispute**, thereby influencing the court simulation and making this feature even more accurate and valuable.

In addition to delivering working software, I'll develop a number of reports, designs and specifications as part of the Major Project process:

- **Developer documentation.** My abstract system will be taking a module of business logic, so I'll need to provide documentation as to how to write a module that can be plugged into the system. I'll either expose an API which can be called directly or trigger events which can be hooked into using a publish-subscribe pattern.
- **Choice of server-side language for the application.** Likely candidates include PHP (the most popular server-side language on the web), Node (for consistency between the back-end and the JavaScript front-end), Ruby on Rails and JavaEE. A short report justifying my decision will be a deliverable I can put in my final report's appendix.
- **Choice of processing language for the business logic modules.** In writing the algorithm that performs the complex logic required, I may choose to use a language well suited to data manipulation, such as Python or R. The algorithm implementation should be separate from the web application and should be relatively easy to execute even from another language, encouraging a nicely decoupled design.
- **Design specification,** including data flow and use case diagrams, to help clarify my final design. This might also include a user interface design.
- **Requirements specification:** though this may be a traditional requirements specification, it is more likely to be a collection of stories which break the project up into manageable chunks, mapped to Cucumber features so that I can develop in a BDD-way.
- **JIRA tickets.** I'll use my own installation of JIRA for project management (managing deadlines, delegating tasks, etc). For code management, including bug fixes, refactoring, and so on, I intend to use GitHub Issues. Both of these systems will provide valuable project history that could be worth including in the appendices of the final report.

Annotated Bibliography

- [1] "Convention for the Unification of Certain Rules of Law with respect to Collisions between Vessels," <http://www.austlii.edu.au/cgi-bin/sinodisp/au/other/dfat/treaties/1930/14.html>, 1910, accessed February 3rd, 2015.

A simpler set of maritime laws; 17 articles only a sentence or two in length.

- [2] "Convention on the International Regulations for Preventing Collisions at Sea, 1972," <http://www.admiraltylawguide.com/conven/collisions1972.html>, 1972, accessed February 3rd, 2015.

The most comprehensive maritime law convention, split into five large sections.

- [3] E. M. S. Agency, "Collision," <http://emsa.europa.eu/marine-casualties-a-incidents/casualties-involving-ships/collision.html>, accessed February 3rd, 2015.

A collection of 100 case studies.

- [4] BlueSeas, "Accident Investigations," <http://www.offshoreblue.com/safety/accident-studies.php>, accessed February 3rd, 2015.

A collection of 19 case studies.

- [5] E. Commission, "Emcod," <https://www.emcod.net/>, 2014, accessed February 3rd, 2015.

Emcod is an open-source online dispute resolution platform, which could form the basis of my Major Project.

- [6] F. A. J. Z. J. N. Davide Carneiro, Paulo Novais, *Online dispute resolution: an artificial intelligence perspective*. Online, 2012, accessed February 3rd, 2015.

Presents a "critical evaluation on the use of Artificial Intelligence (AI) based techniques in ODR", so will be invaluable in deciding how to implement the maritime collision business logic.

- [7] C. J. Giaschi, "Collisions," <http://www.admiraltylaw.com/collisions.php>, accessed February 3rd, 2015.

A collection of over 30 case studies.

- [8] N. Hilborne, "Calls for government to invest in online dispute resolution as part of digital legal services push," <http://www.legalfutures.co.uk/latest-news/calls-for-government-invest-online-dispute-resolution-part-digital-legal-services-push>, 2014, accessed February 3rd, 2015.

A recent article legitimising ODS and giving an idea of where ODS platforms are headed.

- [9] D. Martic, "Online Dispute Resolution for Cloud Computing Services," accessed February 3rd, 2015.

Contains proposals "focusing on e-commerce aspects of dealing with low-cost consumer/seller disputes". Worth reading because it tackles a specific problem (like maritime collisions) - do they handle this concretely or develop an abstract design?