Traveling Strategy

Team members

Name and Student id	GitHub id	Number of story points that member was an **author** on. (Points are only from Traveling Strategy)
Charles Chan 27841523	c-cha16	74
Oumar Ba 40003603	Oumar96	60
S.M.Hassan Karimian 40004904	SMHK341	41
Tyler Delaportas 29720766	TDDEL	55
Steffan Venacious 27192630	svenaci	50
Line Ghanem 27280076	LineG	65
Mikael Samveliant 40003178	MikeSmvl	61
Arminé Iradian 27197144	Armie-i	42

Each group member is responsible for counting their own story points. It is the group leader's duty and responsibility to make sure they are accurate. Please keep in mind that we will check your GitHub stats. Note, if your email and github id are not linked properly you will not be counted properly.

You will lose 1 mark if links below are not clickable.

Project summary (max one paragraph)

Elevator pitch description at a high level

This application is designed for travelers who are seeking information about their travel destination. It provides relevant information such as travel advisory, currency, etc. This application centralizes the data and simplifies the search process for users. This application will be deployed as a web application.

Risk

Describe in bullet points or max one paragraph the most significant risk to the project. Conclude this discussion with how you have attacked this risk first (link to code or stories to provide concrete evidence)

Data gathering

- The importance of having enough data to be able to see valuable information for every country is a risk as some countries might not have government provided statistics. For example, Canada has a branch of the government called Statistics Canada which provides CSV files which could be parsed and displayed on our website. Other countries might not provide data which means we will need to find third party data that might not be as accurate. This leads us to the next risk:

    - On further inspection, finding trusted sources will be difficult for certain countries. (China, japan... countries where english is not the first language) Realistically, this application would target English speaking areas)

Accuracy of data

- As the data is not static, we have to make sure that we fetch up to date data for our users and to make sure that we get our data from reputable sources.

Legal and Ethical issues

Describe in bullet points any legal or ethical issues. If they have been described above in the risks, simply note this.

- Ethically, it is important to provide correct and accurate information, because the user will be relying on it to plan their trip. Providing the wrong travel advisory/emergency contract number can be very costly, especially in a foreign country.
- Legally, the use of some data gathered can be proprietary or protected. While, doing the data gathering part, these issues are verified.

Velocity

(make sure the iteration is **clickable link** to the milestone on github)
Only stories that have stakeholder signoff, demo steps, and tests are counted.

Please tag your released commit using "git tag iteration1"

Project Total: X stories, X points over X weeks

Iteration 1 (4 stories, 28 points) (Link will not work)
Max four sentence paragraph describing main achievements.
The main achievements include setting up the environments for both sub-projects, and basic implementation. The authentication side implemented the basic authentication and a few web pages to experiment with the navbar, and profile settings. The connector side implemented the standalone electron application which will be installed locally at the dental clinics and provide direct access to the database on site.

Iteration 2 (4 stories, 21 points) (Link will not work)
Max four sentence paragraph describing main achievements.
Continuous integration was properly configured for both teams which runs the unit, ui and end to end tests on every commit to verify that both applications (Connector, CRM) are still running as expected. The connector side implemented auto-updates, and linting was added to the connector to ensure that all authors respect the same style of code. Finally, storybook was also added for developing UI components. The authentication team had the task to create owner, administrator, and employee

hierarchy. They also had the task to create user page, functionalities for owners to create roles and be able to set permissions.

Iteration 3, (4 stories, 19 points)
Max four sentence paragraph describing main achievements.
We have moved away from the Arthur.ai project and will be moving forward with Traveling Strategy.

The rest of the sprint was spent setting up the project. More precisely, the frameworks and technologies that we will use were decided, a domain with our project name was purchased, the application's skeleton was written to allow starting development and the application was hosted on a DigitalOcean server. We decided to use an Ubuntu server with Nginx for hosting the application. The right permissions had to be created and the firewall had to be setup correctly to prevent threats. Furthermore, continuous integration with CircleCI was configured to allow our tests to run on every commit. The sprint ended with parsing data to and from a json file that was extracted from a branch of Canada's government, called Statistics Canada.

Release 1 Total: 17 stories, 94 points over 8 weeks

Release 1 aka Iteration 4, (5 stories, 26 points)
Max four sentence paragraph describing main achievements.
Implemented fetching in the frontend via GraphQL resolvers created in the backend. Added Google APIs to be able to have access to autocompletion of regions in our frontend form. Added country comparing components in the frontend and automated tests for these components. These components consist of cards, forms, navbars, containers, etc. The parsing of the data was automated using a python package (HTMLparser). Parsing of other countries to canada (still ongoing/not completed) The parsing of other countries like Australia and New Zealand are going on, but to complete the MVP we decided to push this part to the following sprint and hardcode the attributes needed to do some comparisons.

Iteration 5, (4 stories, 19 points)
Max four sentence paragraph describing main achievements.
Bugs were fixed: mostly UI issues and we needed to refactor the comparing of countries using their ISO value, for less ambiguous country name grabbing. Advisory and visa data from Australia and New Zealand government websites. Language data was parsed for all countries and taken from wikipedia pages. Furthermore, a new package was implemented in the web server to make sure that the application will run again if it goes down unexpectedly or if the server restarts.

Iteration 6, (5 stories, 31 points)
Max four sentence paragraph describing main achievements.

US, Ireland, UK were added as new countries which contain travel advisories and visa information. A new attribute was added and parsed, electrical sockets for all the countries which also includes frequency, and voltage. Additionally, a new attribute of currency for all countries was incorporated; it includes a calculator to convert specific amounts. The final attribute added was time zones for different cities, to facilitate that the comparing of countries, is now done by comparing cities and countries together (Still using country iso).

Iteration 7, (2 stories, 16 points)
Max four sentence paragraph describing main achievements.
Many bugs arose from the original timezone feature, so it needed to be rehauled and moved to the backend. Refactoring was applied to the old parsers, which includes speeding up the parsing, and abstracting logic to make it more reusable. A new attribute of traffic direction was added to show which side of traffic you drive on (left or right). 4 new Central american countries were added, Belize, Mexico, Panama, Dominica and Dominican republic was added. All the visa information is included, but only Mexico had travel advisory data through a link.

Release 2 Total: 19 stories, 112 points over 10 weeks
Release 2, Iteration 8, (8 stories, 46 points)
Max four sentence paragraph describing main achievements.
Additional information was added to explain the advisory in more depth; Carribean countries, Singapore, and Mauritius were added as countries to search from. Unsafe areas was added as an attribute to describe areas that might be deemed unsafe. (Canada was used as a source for this info) Drug legality was added as an attribute that outlines which drugs are legal. Emergency contact/embassy information and health/vaccines attributes were added.

Iteration 9, (X stories, X points)
Max four sentence paragraph describing main achievements.

Iteration 10, (X stories, X points)
Max four sentence paragraph describing main achievements.

Iteration 11, (X stories, X points)
Max four sentence paragraph describing main achievements.

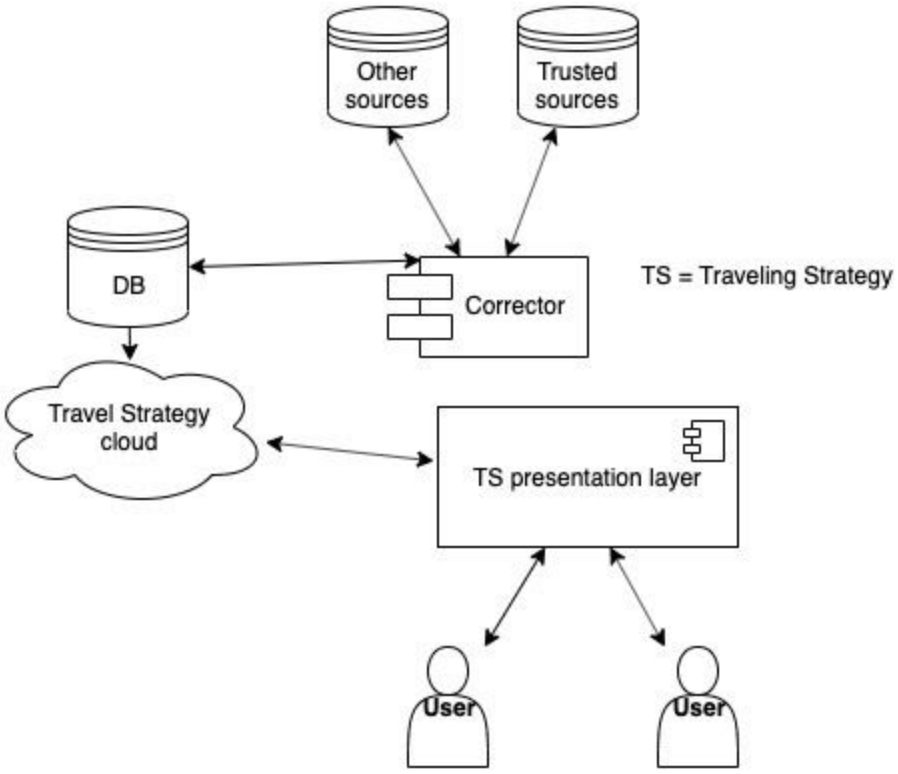
Iteration 12, (X stories, X points)
Max four sentence paragraph describing main achievements.

Release 3 Total: X stories, X points over X weeks
Release 3, Iteration 13, (X stories, X points)
Max four sentence paragraph describing main achievements.

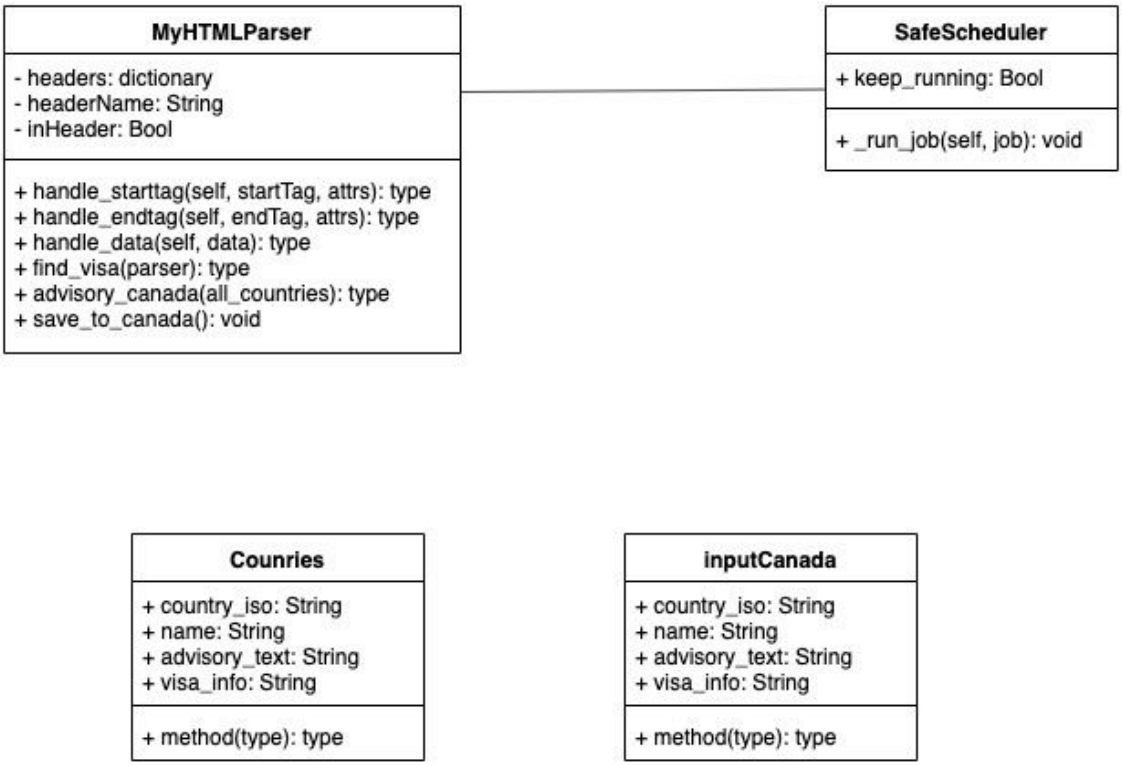
Overall Arch and Class diagram

Show us the layers in your system and your domain classes. You can also include individual class diagrams in your stories on GitHub

Put in your highlevel arch diagram (ie the major components)



High level diagram



Class diagram

For all existing arch diagrams, please only describe changes in your arch

Infrastructure

For each library, framework, database, tool, etc

- React (js)
- Apollo (js)
- GraphQL
- SQLite
- Storybook (js)
- Node.js
- CircleCi
- Eslint (js)
- Babel (js)
- SQLite3 (js)
- Nodemon (js)
- Jest (js)
- Html.parser (python)
- Yagmail (python)
- Schedule (python)

Name Conventions

List your naming conventions or just provide a link to the standard ones used online.

Coding Style

Code

Key files: top **5** most important files (full path). We will also be randomly checking the code quality of files. Please let us know if there are parts of the system that are stubs or are a prototype so we grade these accordingly.

File path with clickable GitHub link	Purpose (1 line description)
travelingstrategy/server/data/advisory_ca.py	Parses json, csv or data files and saves them in the DB
travelingstrategy/server/scheduler/script_automator	Automates the parser to run daily and send emails if anything goes wrong
travelingstrategy/client/src/pages/country	Fetches data from the backend and properly displays it for users
travelingstrategy/client/src/components/CountrySelector	Handles user city or country searches by offering dropdown suggestions and parses the country from the choice for later use
travelingstrateegy/server/src/resolvers/canadaMutations.js	Resolver to abstract database calls easier use in the frontend

Testing and Continuous Integration

Each story needs a tests before it is complete. If some class/methods are missing unit tests, please describe why and how you are checking their quality. Please describe any unusual aspects of your testing approach.

List the **5** most important test with links below.

Test File path with clickable GitHub link	What is it testing (1 line description)
travelingstrategy/client/src/components/Card/Card.stories.js	Example of a story book file that automates UI tests for the frontend
travelingstrategy/client/src/__tests__/App.test.js	Checks that it renders without crashing
travelingstrategy/client/src/__tests__/Header.test.js	Verifies initial state of the props and upon changes it is updated properly
travelingstrategy/server/src/__tests__/resolvers.queries.js	It queries Canada to whatever country that has been chosen.
travelingstrategy/server/src/__tests__/resolvers.mutations.js	Adds countries from canada to that country.

Describe your continuous integration environment. Include a link to your CI.

Our continuous integration is CircleCi, it is used for testing UI and unit testing. Additionally, there is no downtime for deployment