Requirement Specification

L02, Team3
Jiwoo Lee : leej229
Cynthia Liu: liuy363
Jeehyun Yoon: yoonj13
Jennie Li: lij416
Zoe Ning: ningh4

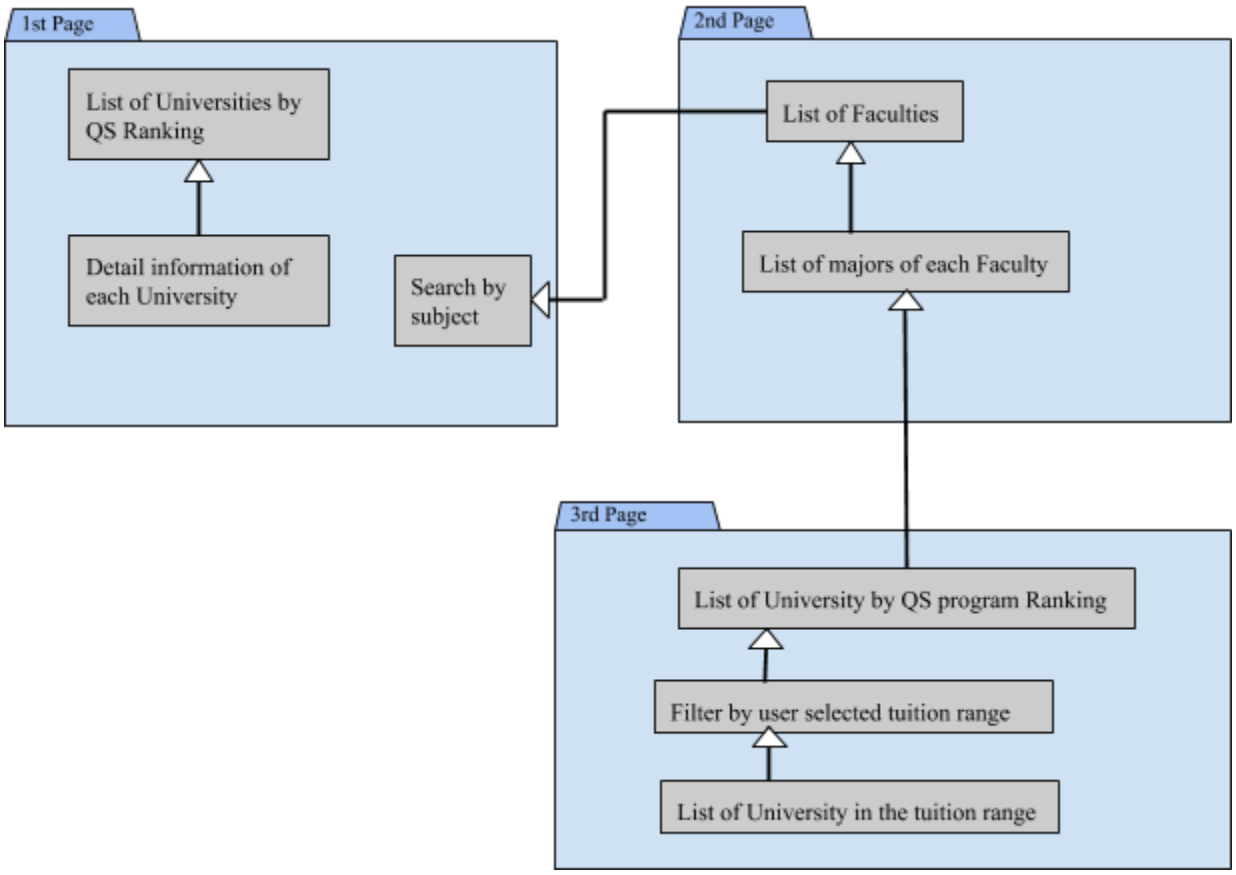
(1) Domain:

Stakeholders	**Explanation**	**Expectations**	**Goals**
High school students (local)	This is our first obvious stakeholder, as high school students will be the main user of our software. The functionality to find, sort, and evaluate different universities and programs are most useful, and will be most used by local high school students (local means they're in Canada).	They expect the software would have admission requirements specific to their case (101). They will also require functions like searching information about specific university programs.	The main goal of our software is to fulfill the needs of the highschool students, especially the 101 audience. When developing our software, the useability and reliability must be focused on them.
High school students (international)	Another major stakeholder of our program would be international students interested in applying to universities in Ontario.	Same expectations as local highschool students, except their admission requirements will be different: International - 105 Local - 101	Same goal as local highschool students, except the database used for developing the functionalities should be specific to the 105 audience.
Parents	For some parents, they would want to know about the universities and give suggestions to their children. An important reason for that is that most of them pay the tuition and they might want to know which one is affordable.	The expectation is basically the same as high school students. The focus for parents might be different. They might care about more realistic problems like the employment rate, tuition and so on.	Same goal as high school students.
University institutions	Minor stakeholder of our software. They're not our main	In some ways this software could be used	One of the plans we have for our software is

	audience but they still might, to gather information about their competitor universities.	as an advertising tool for universities, so they will be interested in how their universities are ranked by the databases we have.	to provide a function where universities are ranked by user keywords. This doesn't directly help universities fulfill their expectations of advertising their university students, but if their ranking is high, it could help towards advertising that university.

(2) Functional Requirements:

- **Initial page** will display a list of universities in Ontario by QS Ranking. When a user selects one university, it'll give them general information about that university. There will be a button to navigate to "rank by subjects" page.
- **Second page** will categorize university majors by their faculty. Each major will be a button to navigate to the third page, with input keyword major="selectedButton.text"
- **Third page** will display university rankings by the major that the user inputs initially in the second page.. Then, the user will be able to select a bunch of filter options (tuition, employment rate, etc) to customize the rankings. Again, when user selects one of the university in the ranking list, it'll show related information about that program, such as admission average and target enrolment



(3) Non-functional requirements:

Qualities	**Answer**
Reliability	Reliability means that it usually does what it's intended to do. The database used in our program must be reliable. We can ensure reliability by gathering data from accredited sources, and data that corresponds to the current year.
Availability	Availability is the ability to answer to requests/be accessible. For our software, the only external variable we're using is the input data files in csv. When writing our code, we should make sure we use relative path rather than absolute path so that data files would still be available in other computers.
Security	Security is a key part of a software. Our design currently doesn't have an authentication system/data gathering for the backend side, but if we ever do, we would want to hash

	user passwords to improve personal security.
Safety	All data in software must not be changed by other customers or people except for developers. Since our program is an information application, credibility is the factor that we will be evaluated. Therefore, we should keep providing accurate and reliable data to clients. If the information is easily modified, the customers will doubt the safety of the application. Hence, we should monitor the dataset continuously and prevent access to data.
Accuracy of result	Accuracy of the results is the most significant aspect. Our main target is secondary school students who prepared to apply to universities. By providing accurate information, we can help customers search for several suitable universities for each customer.
Performance	There are many qualities for evaluating the software's performance. Efficiency and accessibility are the most vital parts of this software. To be specific, we should provide accurate information in a short time to customers. These internal qualities are determined by which algorithms we choose. Therefore, we should desire and design the best efficiency and accessibility.
Human-computer Interface Issues	Our main Human-Computer interface issue is accessibility. We should provide the information to customers whenever and wherever they want. In other words, we have to make sure that customers can access the app even when it is offline.
Operating constraints	Since our program will be designed as a search application, it does not require a complicated operation method. However, there will be an operation constraint in the searching system. Since all information will be written in English, foreign languages can cause unexpected errors. Therefore, there will be a restriction when people use other languages.
Physical constraints	There will be a barrier constraint. For example, when users search with incorrect spelling, the program will suggest alternative options to customers. This function will help clients find suitable universities or programs.
Portability issues	Portability issues are related to the Human-Computer interface issue. Since our software is a mobile application,

	all people can carry their cell phone everywhere except for certain places or special cases. As we mentioned above, the application will operate in an offline environment. Therefore, portability is not the main issue.

(4) Requirements on the development and maintenance process:
Quality Control Procedure:

- The external qualities, as documented in our design specifications, will be checked using unit testing.
- The internal qualities, such as completeness, consistent, modifiable, traceful, unambiguous, correct, verifiable, abstract is what our group members should each keep when they're writing their part of the program.

Likely Changes:
The likely changes of this software would be the ranking over time. QS World Ranking, for example, changes every year so we would need to take this into account and anticipate for change. In our program, we design it to read data from input file so as long as we name the new file the same and replace it, there won't be any problem.

Maintenance:

- Employ maintenance planning and scheduling (check for data change in future years)
- Detect problems early - be smart about unit testing so that we could work towards 0 failure
- Write detailed requirement specification so that if we ever find error we can easily go back and check what that function's really supposed to do (anticipate for change)
- Modularize our program so that if there's any error we can go back to that module only, without having to test other modules (separation of concern)