

precedented :: Vishwaa Sofat, Kazi Jamal, Eric Lau, Raymond Lee, and Ahmed Sultan
SoftDevI pd9
P05 — Fin
2020-06-11

caerus by precededented

Roster and Roles

- **PM Vishwaa Sofat:**
 - Project management
 - Set milestones, and track progress of team
 - Updating design doc
 - Devlog maintenance
- **Kazi Jamal:**
 - Opportunity bulletin
 - Scholarship finder
- **Eric “Morty” Lau:**
 - OAuth
 - Accounts
 - Reminders/notifications
- **Raymond “ray. lee.” Lee:**
 - Opportunity bulletin
 - Scholarship finder
- **Ahmed Sultan:**
 - UI/UX
 - Routing
 - Form submission
 - Productivity

Project Description

caerus is a website based off of Mr. Blumm's weekly Opportunity Bulletin, where students at Stuyvesant can search through a database of opportunities, internships and scholarships curated by Mr. Blumm and the Stuyvesant faculty.

Frontend Framework

We will be using Bootstrap as our frontend framework, as we have all used it extensively and are comfortable with it.

Project Features

Yellow - MVP

Components

- **ACCOUNTS**

- **Administrator accounts**

- Add opportunities/scholarships
 - Add a single opportunity
 - Add multiple opportunities with a CSV file
 - Edit opportunities/scholarships
 - Moderate opportunities/scholarships
 - View/review resumes and applications submitted

- Alumni accounts (*if we have time*)

- Share opportunities/scholarships, to be approved and posted by an administrator
 - Can't view opportunities/scholarships
 - View/review resumes and applications submitted

- **Teacher accounts**

- Add opportunities/scholarships
 - Can view opportunities/scholarships

- **Student accounts**

- View opportunities/scholarships
 - Request opportunities/scholarships
 - Share opportunities/scholarships, to be approved and posted by an administrator (*if we have time*)
 - Perhaps a link to Google Form to suggest new opportunities

- *Managed via OAuth (admin, teacher, student)/custom (alum)*

- **OPPORTUNITY BULLETIN**

- Search/filter/sort by...

- Tags, deadlines/date, requirements, etc.

- Pagination

- Allow students to 'save' opportunities they are interested in

- Allow administrators, teachers, alumni to post opportunities for students

- **Individual opportunities**

- Basic information about the opportunity: deadlines, requirements, etc.
 - Student feedback section about process and experience
 - Extra features
 - How many people have applied in the past?
 - How many have been accepted? etc.

- **SCHOLARSHIP FINDER**

- Search/filter/sort by...

- Tags, deadlines/date, requirements, etc.

- Allow students to 'save' scholarships they are interested in

- Allow administrators, teachers, alumni to post scholarships for students

- **Individual scholarships**

- Basic information about the scholarship: deadlines, requirements, amount of \$, etc.
 - Student feedback section about process and experience

- Extra features
 - How many people have applied in the past?
 - How many have been accepted? etc.

● **PRODUCTIVITY**

- Save opportunities/scholarships (kind of like Pocket/Trello)
 - To-do lists can be created for each opportunity/scholarship to remain on track
(if we have time)

● **REMINDERS/NOTIFICATIONS**

- Automated emails
 - Provide users with reminders for deadlines of opportunities/scholarships they have saved
 - Provide users with notifications when new opportunities with specified 'tags' are released
 - Customized 'weekly update' of things added throughout the week
 - Users can select what filtered tags they want to be sent

● **RESOURCES**

- Links and descriptions for helpful resources

● **STUDENT COLLABORATION**

- Student-to-student/peer-to-peer forum for collaboration, to get teams for projects, and organizations

● **RESUME/APPLICATION REVIEW**

- Can make posts for feedback from alumni/administrators

Database Schemas

Users

- User
 - userID : Integer primary_key
 - token : String
 - email: String
 - name: String
 - accountType: String

Opportunities

- Opportunity
 - opportunityID: Integer primary_key
 - title: String
 - description : String
 - field: String
 - gender: String or NULL
 - location: String or NULL
 - startDate: DateTime or NULL
 - endDate: DateTime or NULL
 - deadline: DateTime or NULL
 - cost: Float
 - datePosted: DateTime
- OpportunityGrade
 - opportunityID: Integer

- grade: Integer
- OpportunityLink
 - opportunityID: Integer
 - link: String

Scholarships

- Scholarship
 - scholarshipID: Integer primary_key
 - title : String
 - description : String
 - deadline: DateTime or NULL
 - eligibility: String or NULL
 - datePosted: DateTime
- ScholarshipLink
 - scholarshipID: Integer
 - link: String

Saved

- SavedOpportunity
 - userID: Integer
 - opportunityID: Integer
 - reminderDate: DateTime or NULL
- SavedScholarship
 - userID: Integer
 - scholarshipID: Integer
 - reminderDate: DateTime or NULL

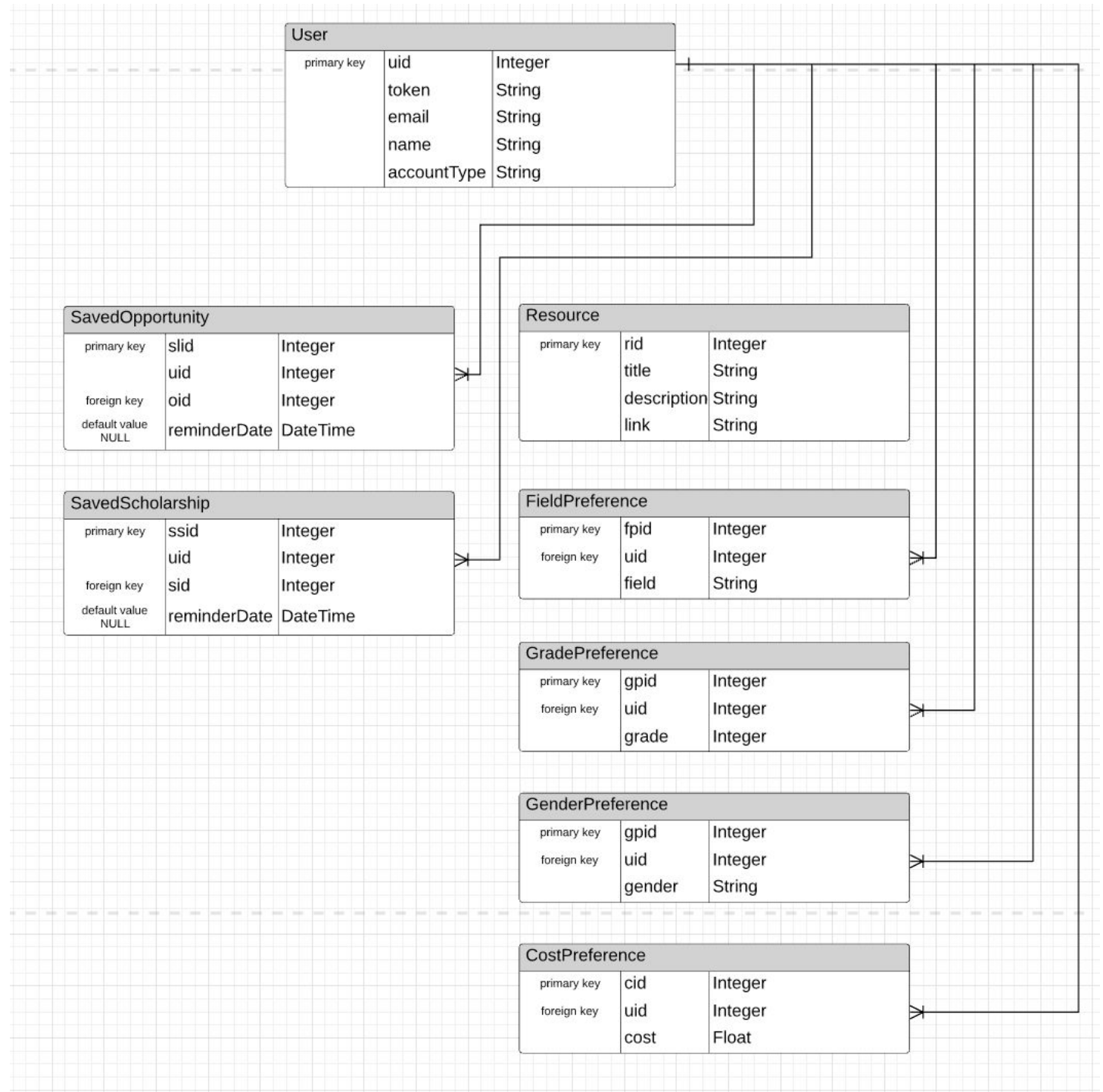
Resources

- Resource
 - resourceID: Integer primary_key
 - title: String
 - description : String
 - link: String

Preference

- FieldPreference
 - userID: Integer
 - field: String
- GradePreference
 - userID: Integer
 - grade: Integer
- GenderPreference
 - userID: Integer
 - gender: String
- CostPreference
 - userID: Integer
 - cost: Float

<https://app.lucidchart.com/invitations/accept/d660ed01-9c0c-4abe-87fd-ef794ba29dfe>



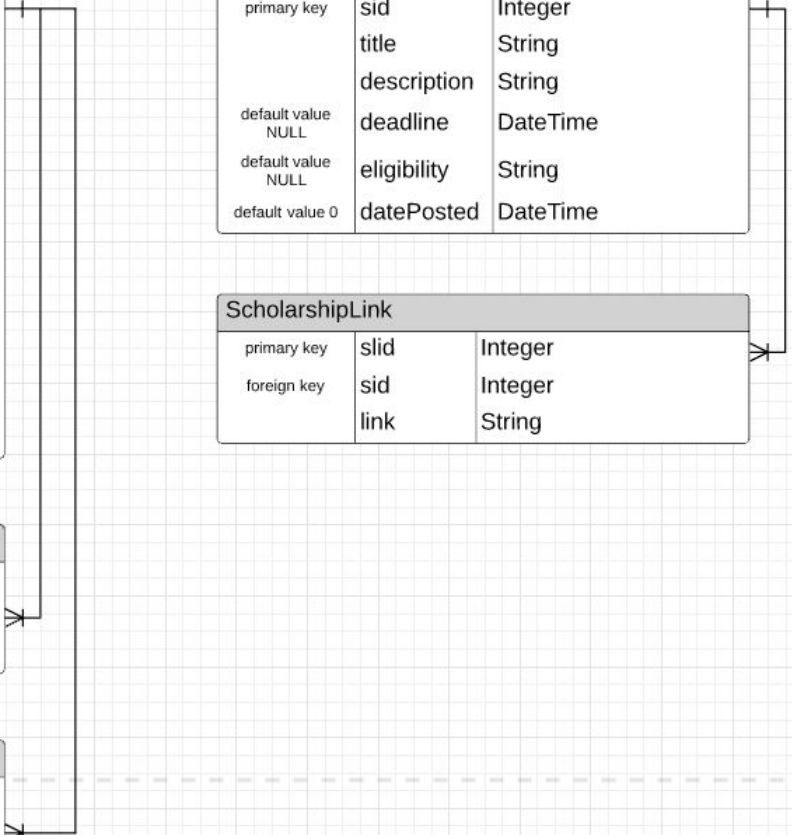
Opportunity		
primary key	oid	Integer
	title	String
	description	String
	field	String
default value NULL	gender	String
default value NULL	location	String
default value NULL	startDate	DateTime
default value NULL	endDate	DateTime
default value NULL	deadline	DateTime
	cost	Float
default value DateTime.now	datePosted	DateTime

OpportunityGrade		
primary key	ogid	Integer
foreign key	oid	Integer
	grade	Integer

OpportunityLink		
primary key	olid	Integer
foreign key	oid	Integer
	link	String

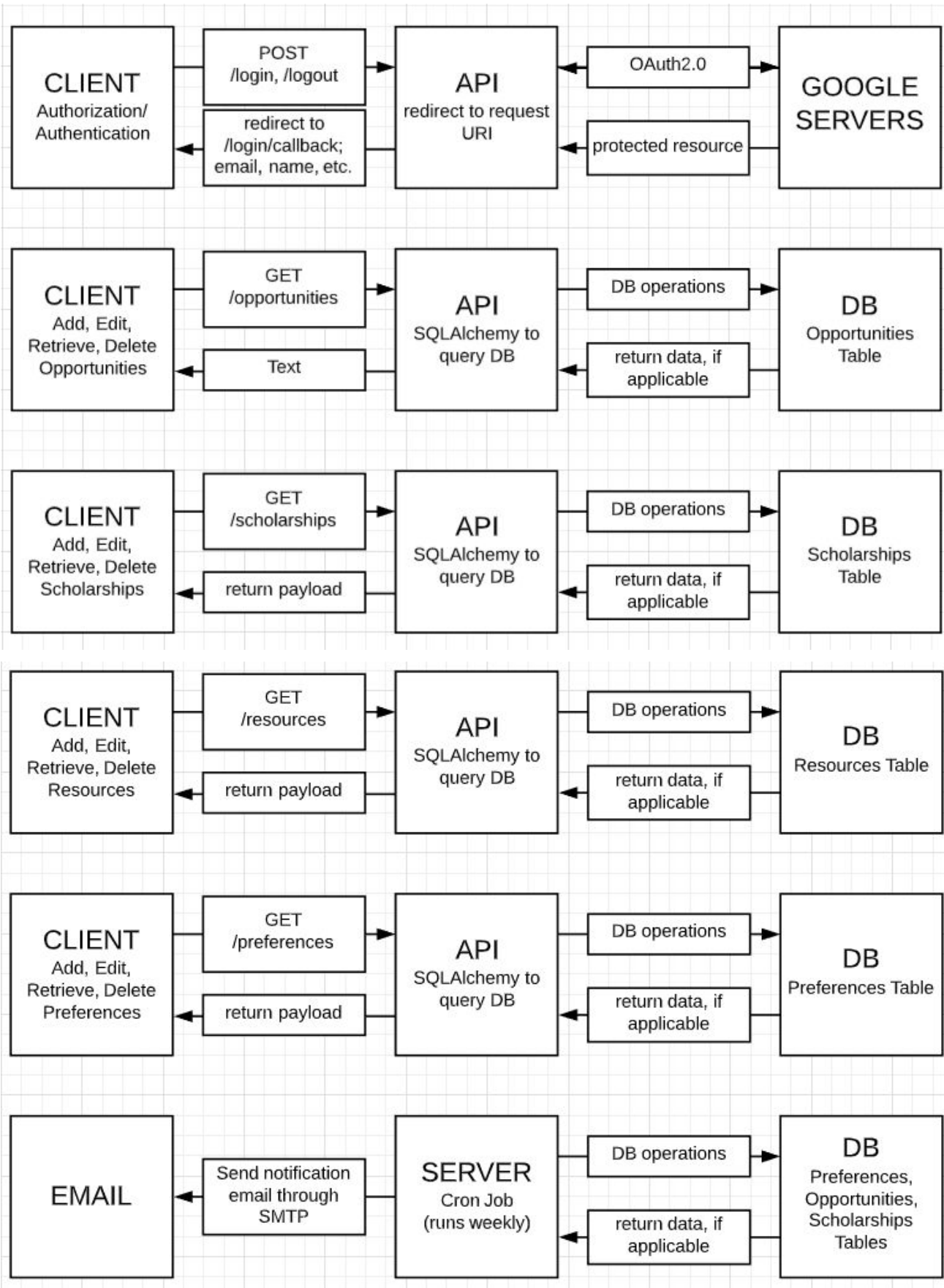
Scholarship		
primary key	sid	Integer
	title	String
	description	String
default value NULL	deadline	DateTime
default value NULL	eligibility	String
default value 0	datePosted	DateTime

ScholarshipLink		
primary key	slid	Integer
foreign key	sid	Integer
	link	String

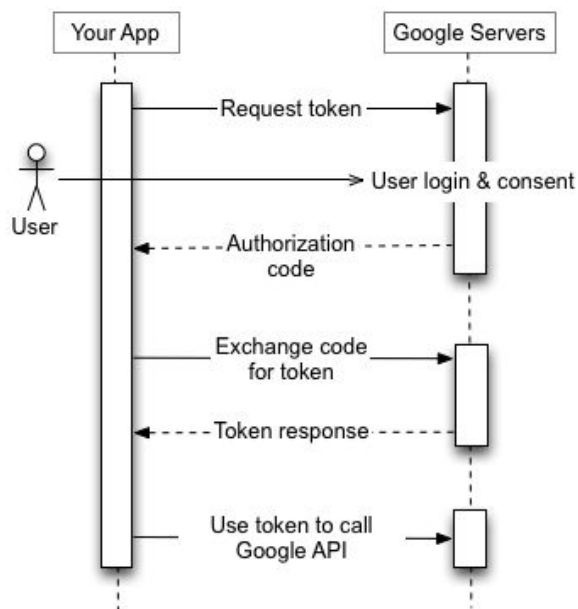


Component Map

<https://app.lucidchart.com/invitations/accept/2332b742-e4fd-4d7a-94c8-712c206354d5>



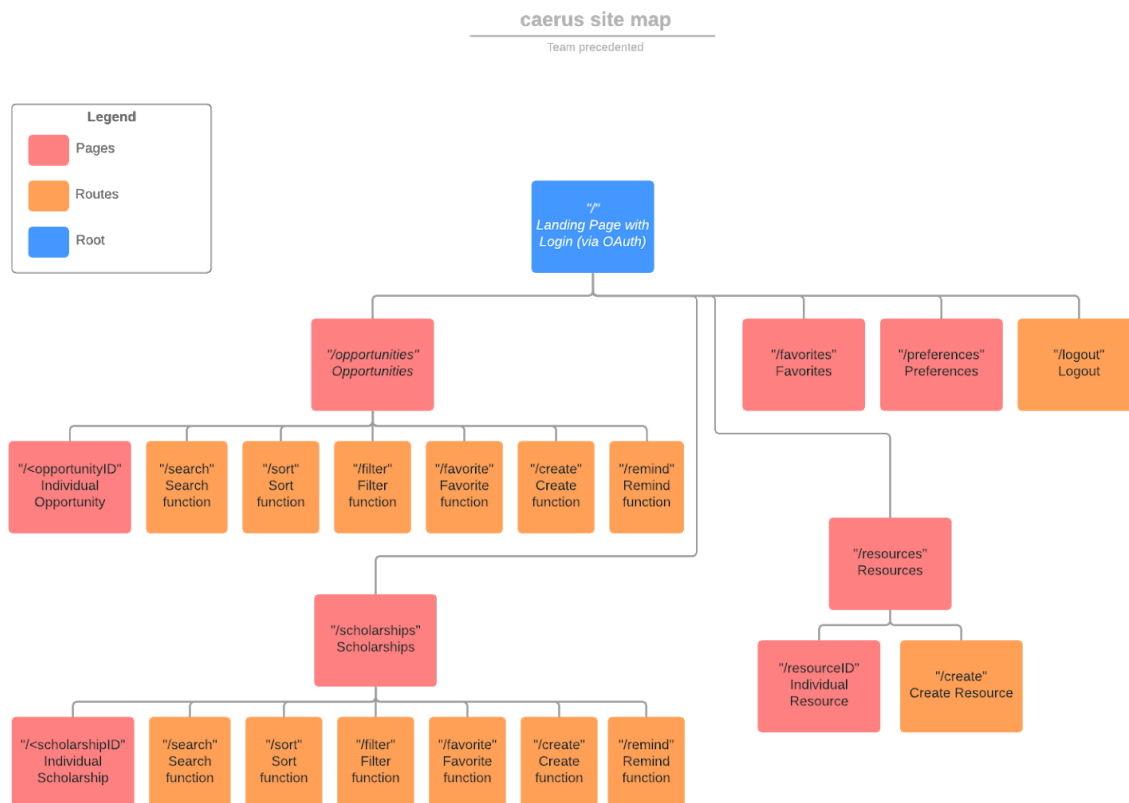
How OAuth2.0 will work behind the scenes:



<https://developers.google.com/identity/protocols/oauth2>

Site Map

<https://app.lucidchart.com/invitations/accept/0e84ea7f-ad71-4583-b985-818f9a771d88>



GET /

GET /opportunities

GET /<opportunityID>

POST /search

POST /sort

POST /filter

POST /favorite

POST /create

POST /remind

GET /scholarships

GET /<scholarshipID>

POST /search

POST /sort

POST /filter

POST /favorite

POST /create

POST /remind

GET /favorites

GET /resources

GET /<resourceID>

POST /create

GET /preferences

POST /logout