Up-to-Date Publication Embedding Iteration 1

Bikram Khanal, Korn Sooksatra and Alibek Zhakubayev

November 10, 2021

1 Project Vision

The purpose of this paper is to automate the publication of the professor up-to-date using JavaScript. It will develop an embedded JavaScript library (similar to Twitter/Facebook timeline embedding) that can be customized. The Up-to-Date publication embedding application will provide features to the user like:

1. List all the available publications of the Baylor University professor.
2. Group the publication by department.
3. List the publication sorted by publication year, or by number of citation.
4. List the publication for certain year or with certain number of citations.
5. Add/delete the items in do not publish list.

2 Requirements

2.1 Functional Requirements

1. A faculty should be able to register to the system.
2. A faculty can obtain a Javascript code or a URL that generates Json or HTML.
3. A reader can filter by some columns.
4. A faculty can correct his/her own publication list.
5. A reader can obtain citations for any publication.

2.2 Non-functional Requirements

1. A faculty can still show his/her publication list on his/her web page although our server is down.
2. All of the publications should be updated every month.

2.3 Business Rules

1. Only faculty can request individual JavaScript, Json or HTML.
2. A faculty can be a user.
3. Only admin can update do not publish list.
4. Only a faculty can correct his/her own publication list.

3 Use Cases

3.1 Use Case Diagram

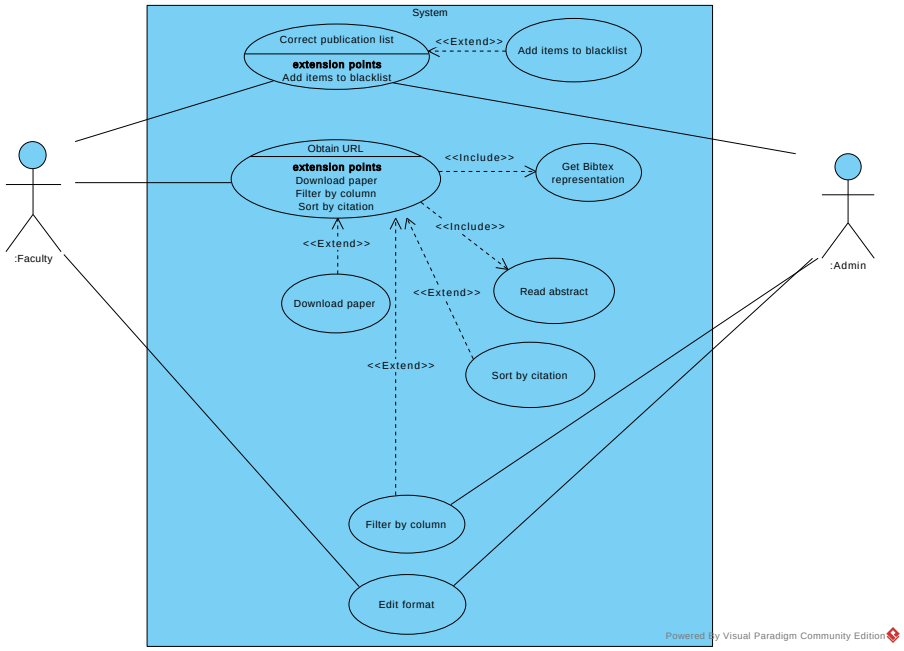


Figure 1: Use case diagram.

Figure 1 shows the use case diagram of our project. Further, each of the use cases are explained in detail in the following sub section.

3.2 Read abstract

Use case UC1: Read the abstract of the paper.
Scope: Publication listing of an individual on their website.
Level: User goal.
Primary actor: Students, professors, scholars(users).
Stakeholders and Interest: Users want to read the abstract of the paper published by professor "A".
Precondition:

- The user has access to a web browser and the internet.
- The user can read and understand English.
- User knows the title of the paper.
- User knows the publication year(Optional).

Success Guarantee (or Post Conditions): User will read the abstract of the paper he is looking for. **MainSuccess Scenario(or basic flow)**:

1. User wants to read an abstract of the paper "Title".
2. User navigates to the homepage of the professor(system).
3. The system presents options of teaching, schedules, Bio, and publication to the user.
4. User selects publication.
5. The system lists the publication of professor "A" sorted by date in descending order.
6. Users search the title of the paper(using keyboard) or surf the page until the desirable listing is located.
7. The system provides an option of "access online" at the end of each publication listing.
8. User makes his selection.
9. The system navigates the user to the online library.
10. The system displays the abstract of a paper on the screen.
11. Users can read the abstract of the paper.

Extensions:
2. a The homepage of Professor "A" is down
   1. The system throws a "404 page not found" error.

6.a The publication is not listed on the system.

1. Users can request an admin to sync the publication.
2. Users can provide his/her email address and be on the mailing list to be informed when the website will sync next time.

3.3

Use Case UC2: Download the latest published paper by professor "A".
Scope: Publication listing of an individual on their website.
Level: User goal.
Primary Actor: Students, professors, or any website visitors(User).
Stakeholders and Interest: Website visitors(in most cases, the students or the professor) want to download the latest publication of professor "A" with minimal effort.
Precondition:

- The user knows how to use computers and the internet.
- The user has access to a web browser and the internet.
- The user can read and understand English.
- The user might need access to online libraries.

Success Guarantee (or Post Conditions): The user is ready to download the latest paper of professor "A."
Main Success Scenario(or basic flow):

1. The user wants to download the latest publication of professor "A."
2. The user navigates to the homepage of professor "A"(System).
3. The system presents options of teaching, schedules, Bio, and publication to the user.
4. User selects a publication.
5. The system lists the publication of professor "A" sorted by date in descending order.
6. The system provides an option of "pdf download" at the end of each publication listing.
7. Users can select this option and download the paper.

Extensions:
2. a The homepage of Professor "A" is down.
The system throws a "404 page not found" error.
5. a The listing of the publication is not synced lately, so the latest publication on the website is the second latest publication.

1. Users can request the admin to sync the publication.
2. Users can provide his/her email address and be on the mailing list to be informed when the website will sync next time.

7. a The download options navigate the user to the online library.

1. Users can log in with his/her credentials and reach step 7.
2. Users can request the library for pdf access.

3.4 Add to do not publish list.

Use Case UC3: add publication on do not publish list.
Scope: Publication listing of an individual on their website.
Level: User Goal.
Primary Actor: Admin, professor(user). **Stakeholders and Interests:** Professor or admin does not want certain papers to be listed on the website.
Precondition:

- The user has access to a web browser and the internet.
- User knows the title of the paper.
- Users' credentials have an admin privilege.

Success Guarantee (or Post Conditions): The user will be able to add items on the "Do not publish" list.
Main Success Scenario (or basic flow):

1. User wants to update the do not publish list.
2. The user navigates to the homepage of the developer(System).
3. User login to the system with admin credentials.
4. System presents options of "do not publish list" and "add privilege user".
5. User makes his selection of the do not publish list.
6. System list all items from this list.
7. System presents an option to add or delete paper on this item.
8. User makes his selection of the add item.
9. User provides the title of the publication.
10. System checks the availability of the paper.
11. System asks the user for the confirmation and apply the changes.
12. Users ask the system to save the changes.
13. System does not publish the publication on the website.
14. Users exit the system.

Extensions:
10.a. The paper is not found.

1. Users can still ask the system to add the title on the list.
2. System moves to the next step.

3.5 Obtain URL

Use Case UC4: Obtain a URL that automatically updating a list of publications.
Scope: Publication listing of an individual on their website.
Level: User Goal.
Primary Actor: Professor(user).
Stakeholders and Interests: Professor wants to update his/her list of publications in the web page.
Precondition:

- The user provides some inputs (e.g., his/her email and preferred format).
- Success Guarantee (or Post Conditions): The user has his/her list of publications automatically updated on the web page.

Main Success Scenario (or basic flow):

1. The user accesses our web page.
2. The user goes to the Faculty section.
3. The user fills in some information and chooses which type of format he/she prefers (i.e., Javascript, Json or HTML).
4. The user submits the information.
5. The user obtains a Javascript or URL.
6. The user uses the Javascript or URL to automatically generate and update his/her publication list on his/her web page.
7. The user's list of publications is shown on his/her web page.

Extensions:
6.a. Our server is down.
1. We maintain a cache for each user so that his/her web server does not need to fetch information from our service every time.

3.6 Correct a Publication List

Use Case UC5: Correct a list of publications.
Scope: Publication listing of an individual on their website.
Level: User Goal.
Primary Actor: Professor(user).
Stakeholders and Interests: Professor wants to edit information in his/her list of publications in the web page.
Precondition: The user's web page can show his/her list of publications; however, he/she notices that some information is wrong.
Success Guarantee (or Post Conditions): The user can edit information

of his/her list of publications, and his/her web page shows what he/she has just edited.

Main Success Scenario (or basic flow):

1. The user sends an email to us regarding his/her request to edit some information of his/her list of publications.
2. Our service sends an email back to the user with a link.
3. The user accesses the link to edit some information of his/her list of publications.
4. The user submits the edited information.
5. The user's web page can correctly show his/her list of publications.

Extensions:

1.a. Our server is down.

1.We maintain a cache for each user so that his/her web server does not need to fetch information from our service every time.

3.7 Edit a Format

Use Case UC6: Edit a format of a user's list of publications.
Scope: Publication listing of an individual on their website.
Level: User Goal.
Primary Actor: Professor(user).
Stakeholders and Interests: Professor wants to use his/her own format to show a list of publications and show it in his/her web page.

Precondition: The user's web page can show his/her list of publications; nonetheless, he/she wants to change the format.

Success Guarantee (or Post Conditions): The user's web page can show his/her list of publications in his/her specific format.

Main Success Scenario (or basic flow):

1. The user sends an email to us regarding his/her request to edit the format of his/her list of publications.
2. Our service sends an email back to the user with a link.
3. The user accesses the link to specify the format of his/her list of publications.
4. The user submits the specific format.
5. The user's web page can correctly show his/her list of publications in his/her format.

Extensions:

1.a. Our server is down.

1. We maintain a cache for each user so that his/her web server does not need to fetch information from our service every time.

3.8 Sort by Citations

Use Case UC7: Sort publications by number of citations or year.
Scope: Publication listing of an individual on their website.
Level: User Goal.
Primary Actor: Any user.
Stakeholders and Interests: User might want to read most recent or cited publication.
Precondition: The user opened the list of publications that were published by the Professor.
Success Guarantee (or Post Conditions): The user see ordered list of publications.
Main Success Scenario (or basic flow):

1. The user open our website.
2. The user selects a Professor.
3. The system outputs the list of Professor's publications.
4. The user clicks on year/# of citations column.
5. The system will order the list of citations in ascending order by selected column.
6. The user clicks on selected column again.
7. The system will order the list of citations in descending order by selected column.

Extensions:
7.a. Our server is down.
f 1.We maintain a cache for each user so that his/her web server does not need to fetch information from our service every time.

3.9 Get BibTeX Representation

Use Case UC8: Get BibTeX representation of the publication.
Scope: Publication listing of an individual on their website.
Level: User Goal.
Primary Actor: The user who wants to cite a paper.
Stakeholders and Interests: User wants to cite a paper and get its BibTeX representation.
Success Guarantee (or Post Conditions): The user see ordered list of publications.
Main Success Scenario (or basic flow):

1. The user open our website.
2. The user selects a Professor.

3. The system outputs the list of Professor's publications.

4. The user clicks on cite button near the publication.

5. The system constructs BibTeX representation from the cached information.

6. The system open pop up window with the BibTeX.

Extensions:
7.a. Our server is down.
1.We maintain a cache for each user so that his/her web server does not need to fetch information from our service every time.

3.10 Filter By column

Use Case UC9: Filter publication by any column.
Scope: Publication listing of an individual on their website.
Level: User Goal.
Primary Actor: The user who wants to find papers that have some common feature.
Stakeholders and Interests: User wants to find a list of papers that have the same co-author/ was published in the same year / was published in a particular conference etc.
Success Guarantee (or Post Conditions): The user see filtered list of publications.
Main Success Scenario (or basic flow):

1. The user open our website.

2. The user selects a Professor.

3. The system outputs the list of Professor's publications.

4. The user clicks on arrow down button on the column.

5. The system shows the input area below the column name.

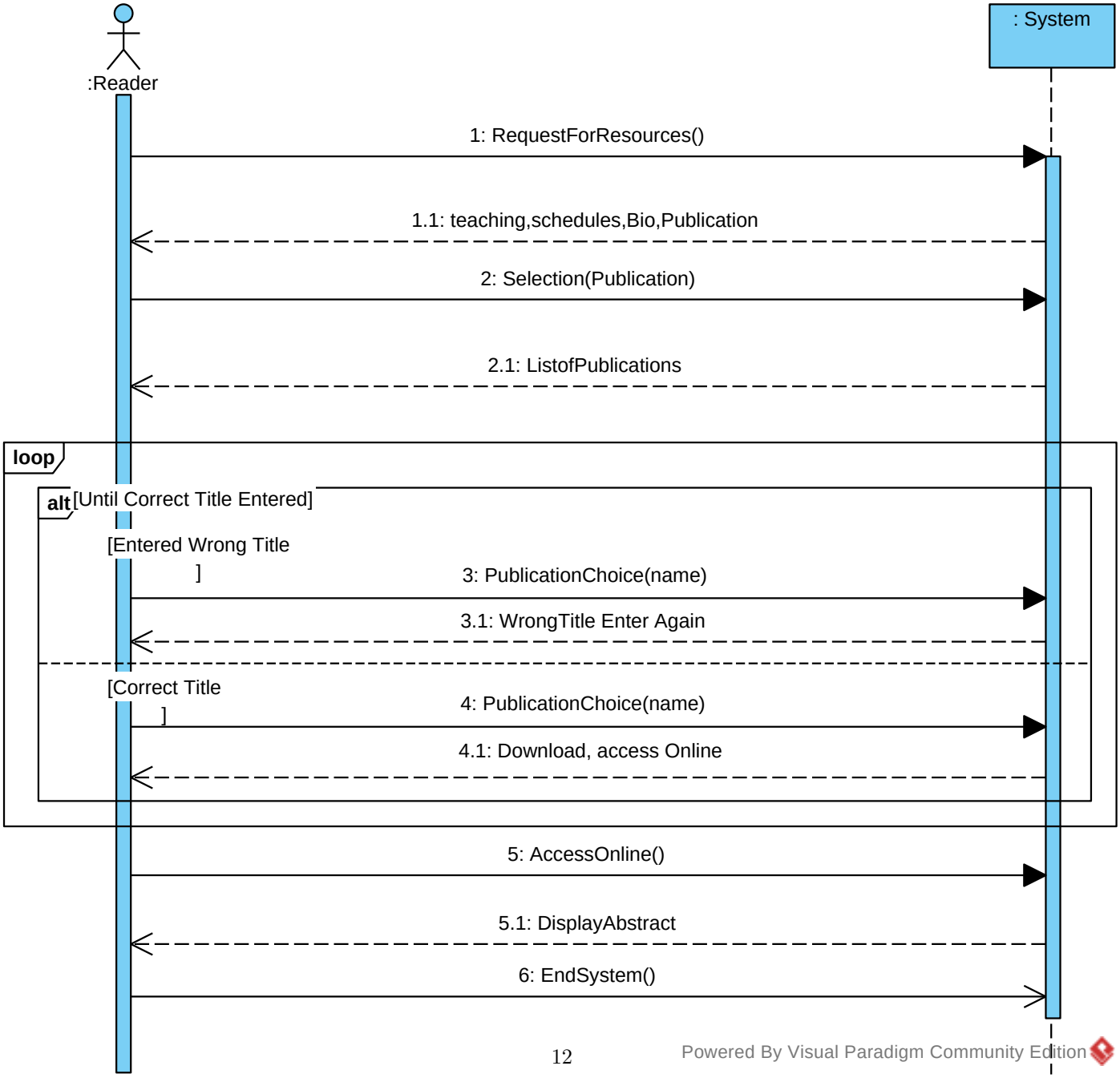
6. The user inputs a string A.

7. The system outputs the list of publications where a particular column contains a substring A.

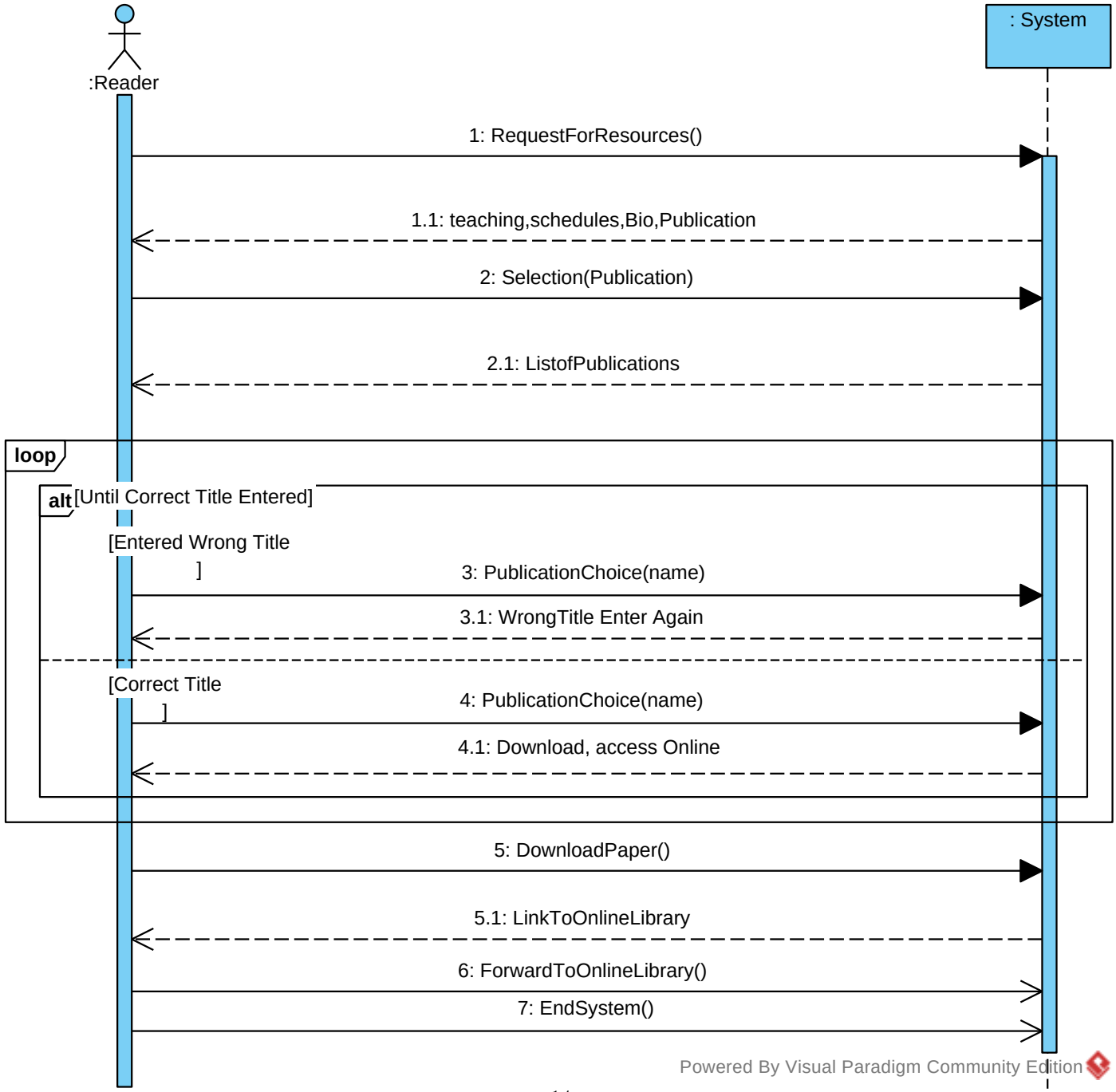
Extensions:
7.a. Our server is down.
1.We maintain a cache for each user so that his/her web server does not need to fetch information from our service every time.

4 System Sequence Diagram

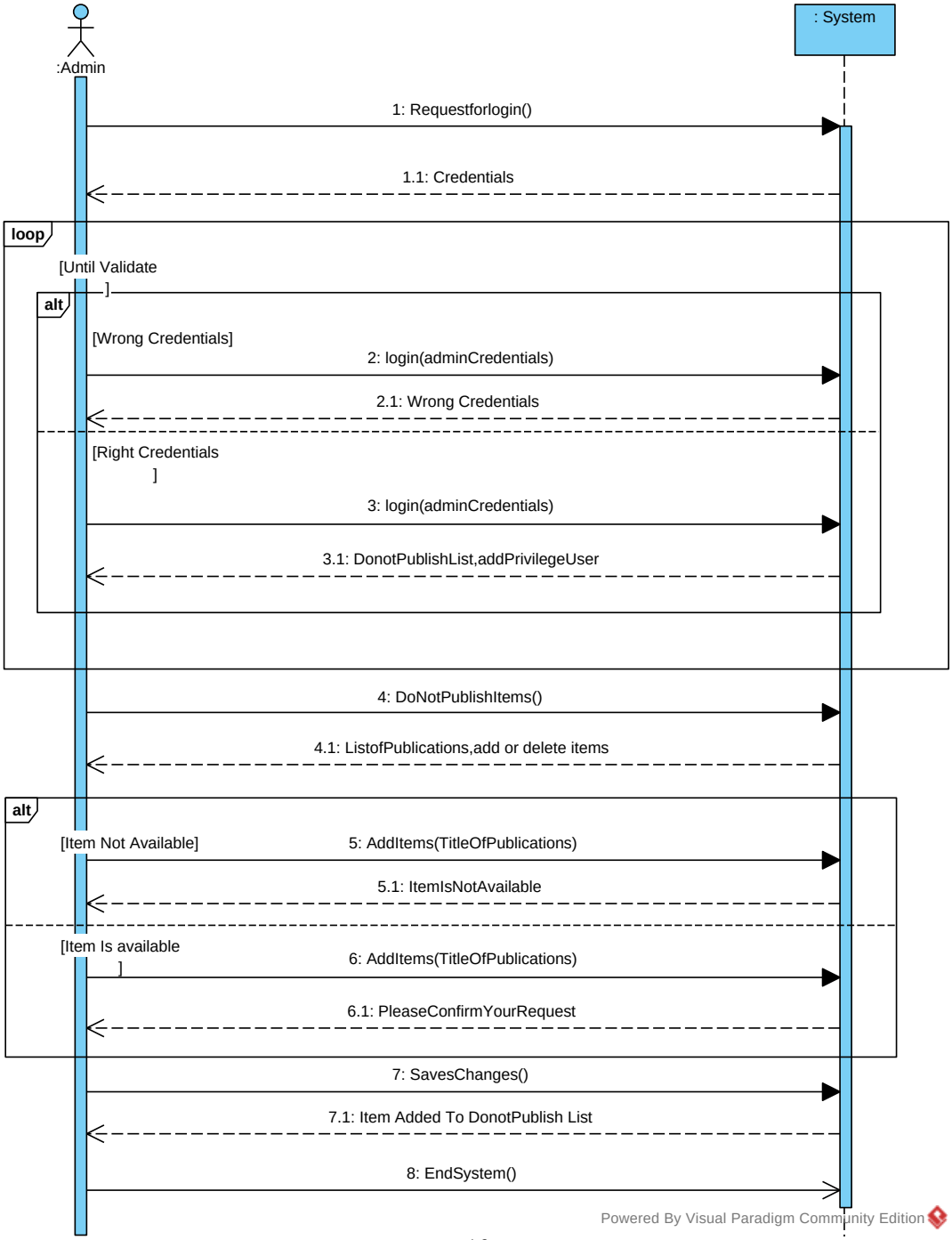
4.1 Read Abstract



4.2 Download Paper



4.3 Add Item to Do not Publish List



4.4 Obtain URL

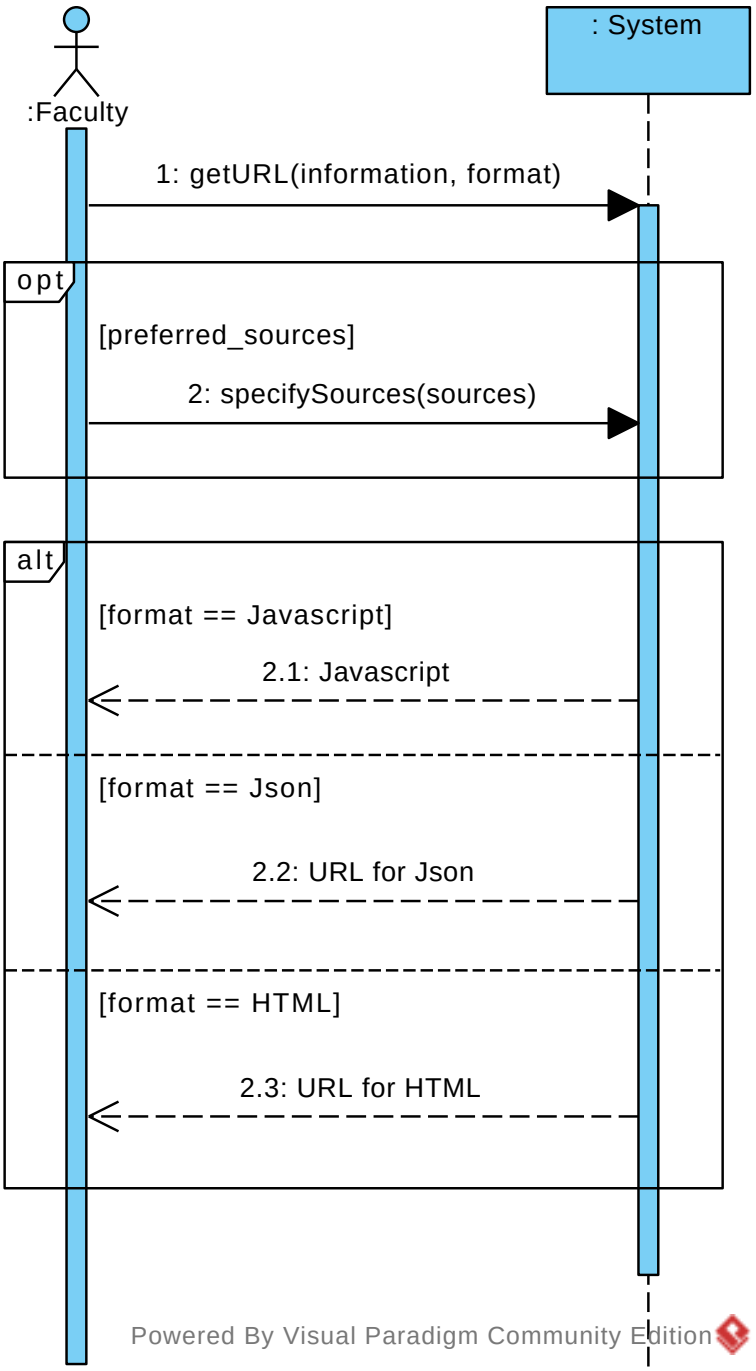


Figure 5: System Sequence diagram to obtain a URL for generating a publication list.

4.5 Correct a publication list

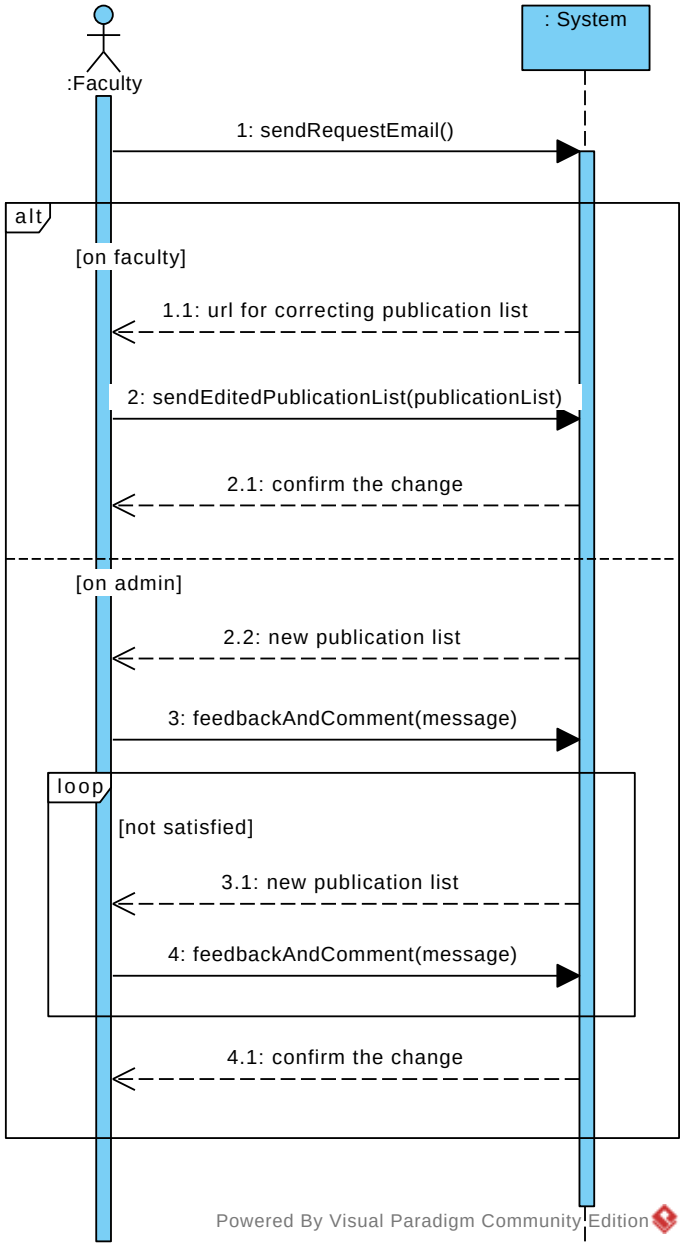


Figure 6: System Sequence diagram to correcting a publication list.

4.6 Edit a format

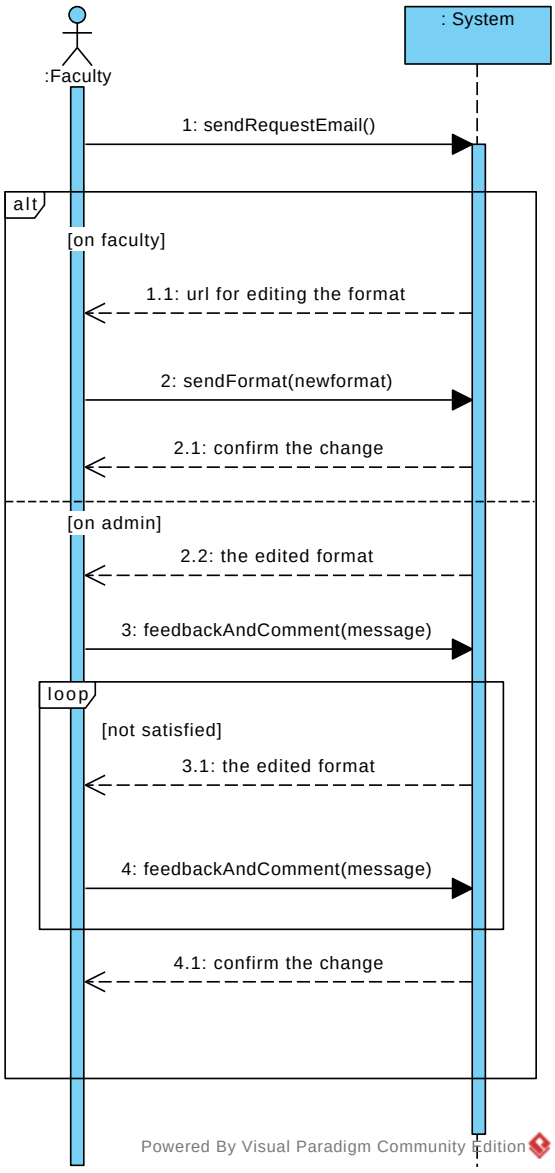


Figure 7: System Sequence diagram to obtain a URL for editing a format of a publication list.

4.7 Sort by Citations

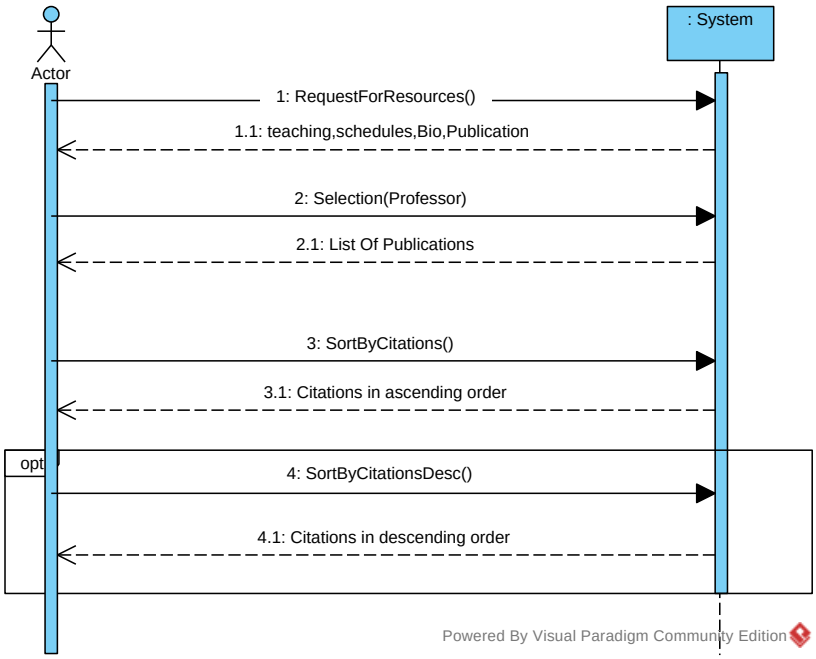


Figure 8: System Sequence diagram to obtain a sorted list of publications

4.8 Get BibTeX Representation

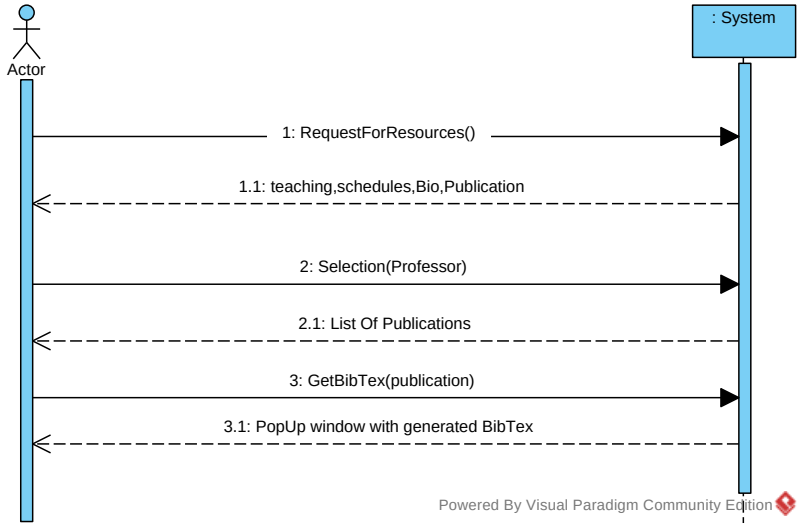


Figure 9: System Sequence diagram to obtain a BibTex representation of publication

4.9 Filter By column

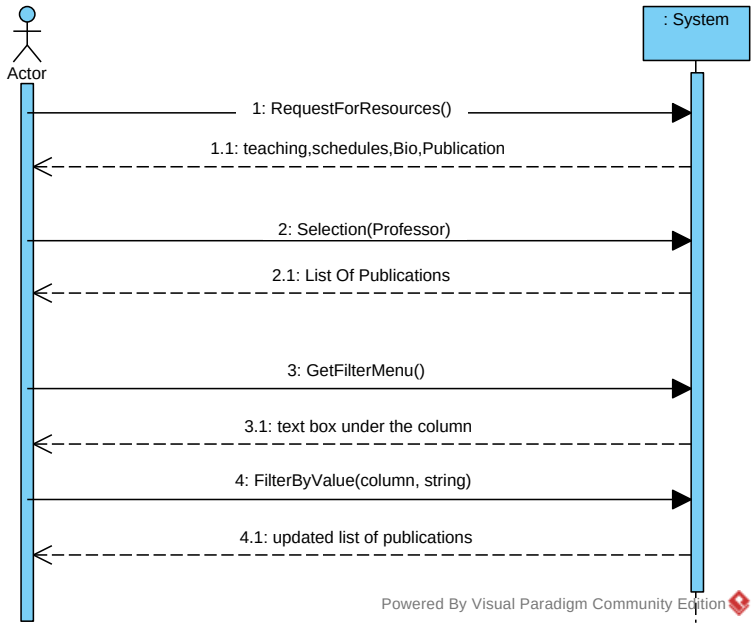


Figure 10: System Sequence diagram to filter list of publications by any column.

5 Traceability Matrix

Use case \ Requirement	1	2	3	4	5
UC1		■			
UC2		■			
UC3					
UC4		■			■
UC5				■	
UC6					
UC7		■			■
UC8		■			
UC9		■	■		

Table 1: Traceability matrix.

6 System Operations

System
requestForResources()
selectionOfProfessoe(professor)
login(username,password)
selection(Publication)
sortByCitation()
SortByCitationDesc()
PublicationChoice(title)
accessOnline()
doNotPublishItems()
AddItems(title)
savechanges()
DownloadPaper(title)
forwaredtoOnlineLibrary()
getURL(information,format)
specifySources(sources)
sendRequestEmail()
sendEditedPublicationList(publicationList)
feedbackAndComment(message)
sendFormat(newFormat)
getBibTex(publication)
getFilterMenu()
FilterByValue(column,string)
endSystem()

Table 2: System Operations for SSDs

7 UML Class Diagram

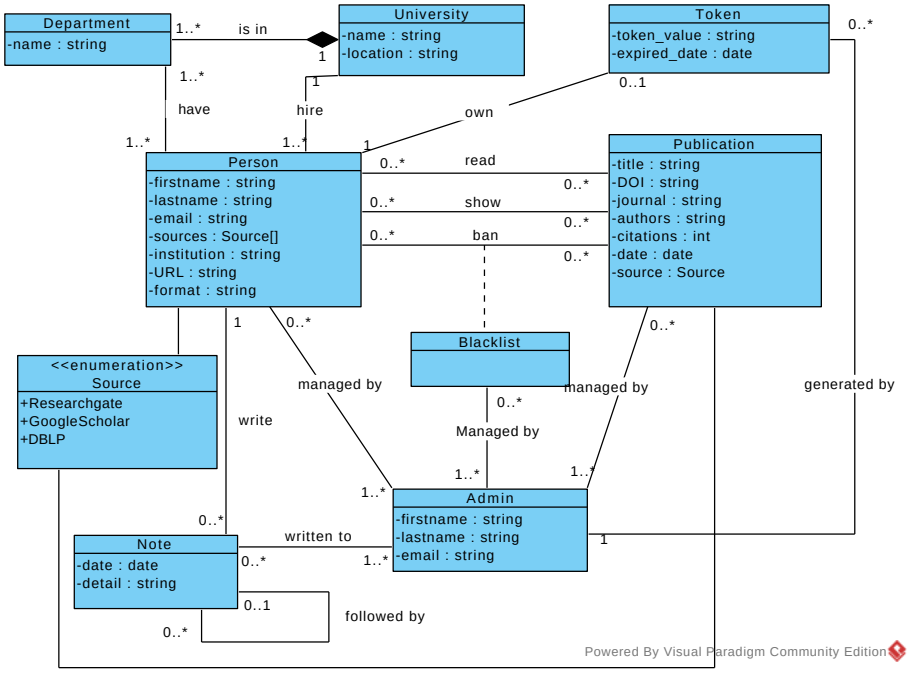


Figure 11: UML class diagram.

Our project mainly consists of four classes as the follows:

1. **Person.** This class can be a reader or a faculty. A faculty would like to use our project to automatically generate his/her publication list on his/her web page. Therefore, this is our main target. Further, a reader would like to search and read some publications.

2. **Publication.** This class is a collection of publications crawled by our project.

3. **Blacklist.** This class is a collection of publications banned by some faculties.

4. **Admin.** This class' responsibility is to manage all the faculties, publications and banned publications (in the blacklist). That is, he/she is able to create, remove or edit those classes.

5. **Note.** This is a note from a faculty to an admin for some request.

Further, the association among those four classes is shown in Figure 11.

8 Operation Contracts

8.1 Read Abstract

Contract C01: readAbstract

Operations: ReadAbstract(titleOfThePublication)
Cross References: Use cases: Read Abstract
PreConditions:
User has requested the publication list.
The publication is available in the system.
PostConditions:
Publication is fetched from publication class.
User is able read an abstract from publication.

8.2 Download Paper

Contract C02: downloadPaper

Operations: downloadPaper(titleOfThePublication)
Cross References: Use cases:Download Paper.
PreConditions:
User has requested the publication list.
The publication is available in the system.
User has an access to download site.
PostConditions:
Publication is fetched from publication class.
System navigate the user to the download page.a

8.3 Add an Item to do not Publish List

Contract C03: DoNotPublishList

Operations: doNotPublishList(PublicationTitle)
Cross References: Use Cases: Add Item to Do not Publish List
PreConditions:
User is an admin.
The publication is on the database.
PostConditions:
The user is able to add an item to the donot publish list.
The publication is not listed on the system.

8.4 Obtain URL

Contract C04: GetURL

Operations: GetURL(information, format)
Cross References: Use Cases: Obtain URL
PreConditions:
A faculty submits his/her information and format.
PostConditions:
Some instances in Publication are fetched according to the information. The faculty instance is updated with an URL in his/her URL attribute. The faculty obtains the URL and can use it to generate his/her publication list.

Contract C05: specifySources

Operations: specifySources(sources)
Cross References: Use Cases: Obtain URL
PreConditions:
A faculty submits his/her information and format.
The faculty specifies which sources he/she would like to refer. **PostConditions:**
The faculty instance's sources attribute is updated with his/her specified sources.

8.5 Correct a publication list

Contract C06: sendRequestEmail

Operations: sendRequestEmail()
Cross References: Use Cases: Correct a publication list and edit a format
PreConditions:
A faculty can use our output to automatically generate his/her publication list.
PostConditions:
An instance of a note is created with the request as its detail.

Contract C07: sendEditedPublicationList

Operations: sendEditedPublicationList(publicationList)
Cross References: Use Cases: Correct a publication list
PreConditions:
A faculty can use our output to automatically generate his/her publication list.
PostConditions:
Instances in Blacklist are created with the publications that no longer are in the publication list and the faculty.
Some instances in Publication are updated with the information in the publication list.

Contract C08: feedbackAndComment

Operations: feedbackAndComment(message)
Cross References: Use Cases: Correct a publication list and edit a format
PreConditions:
A faculty can use our output to automatically generate his/her publication list.
PostConditions:
An instance in Note is created and associated with the request. Its date attribute is filled by the current date, and its detail is filled by the feedback and comment from the faculty.

8.6 Edit a format

Contract C09: sendFormat

Operations: sendFormat(newformat)
Cross References: Use Cases: Edit a format
PreConditions:
A faculty can use our output to automatically generate his/her publication list.
PostConditions:
The format attribute of the faculty instance in Faculty is updated with the new format.

8.7 Sort by Citations

Contract C10: SortByCitations

Operations: SortByCitations()
Cross References: Use Cases: Sort by Citations
PreConditions:
A reader see Professor's list of publication
PostConditions:
Method that is responsible for the data is calling order by function, which will result in correct order of the publications.

8.8 Get BibTeX Representation

Contract C11: getBibTex

Operations: getBibTex(publication)
Cross References: Use Cases: Get BibTeX Representation
PreConditions:
A reader see the publication that he/she want to cite in the list.
PostConditions:
Publication table is accessed, and all needed information is taken from it.
Based on that information, algorithm constructs BibTex citation.

8.9 Filter By column

Contract C12: FilterByValue

Operations: FilterByValue(column, string)
Cross References: Use Cases: Filter By column
PreConditions:
A reader see the list of publications, and want to filter any column by a particular substring
PostConditions:
Service method will filter all rows by comparing if provided substring is part of the column.

9 Activity diagram

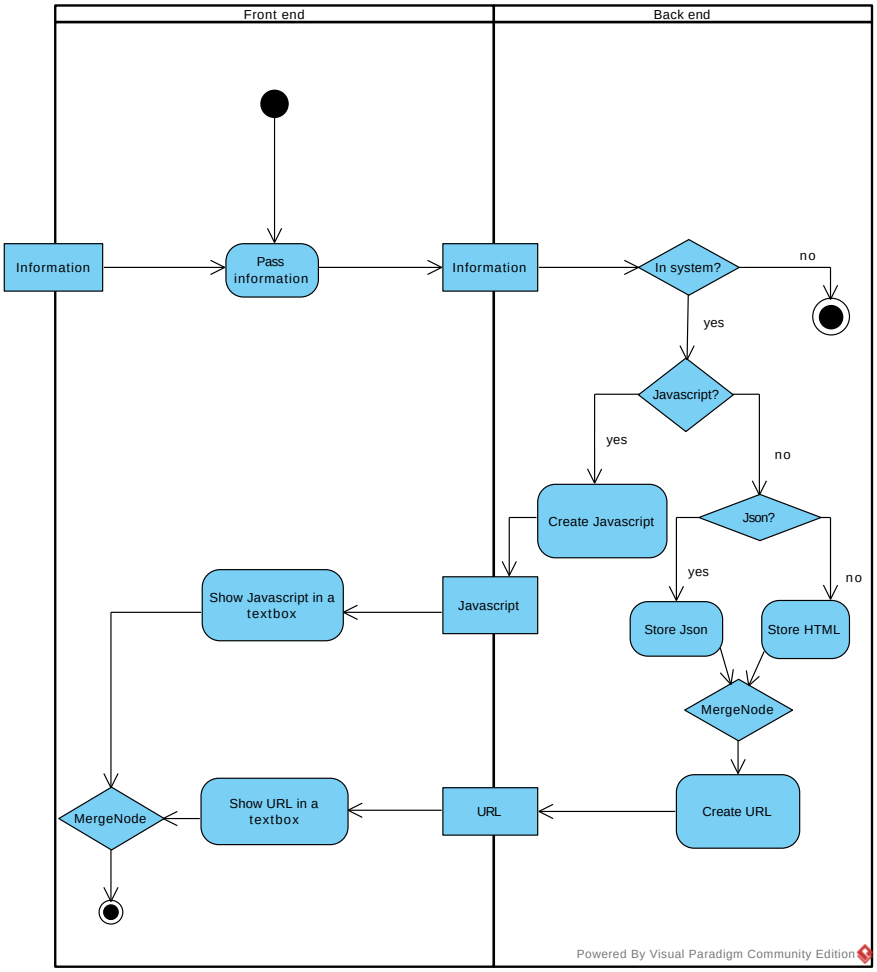


Figure 12: Activity diagram.

Figure 12 shows an activity diagram for processing a request from a faculty to generate a Javascript code, Json or HTML.

10 WireFrames

10.1 Faculty page

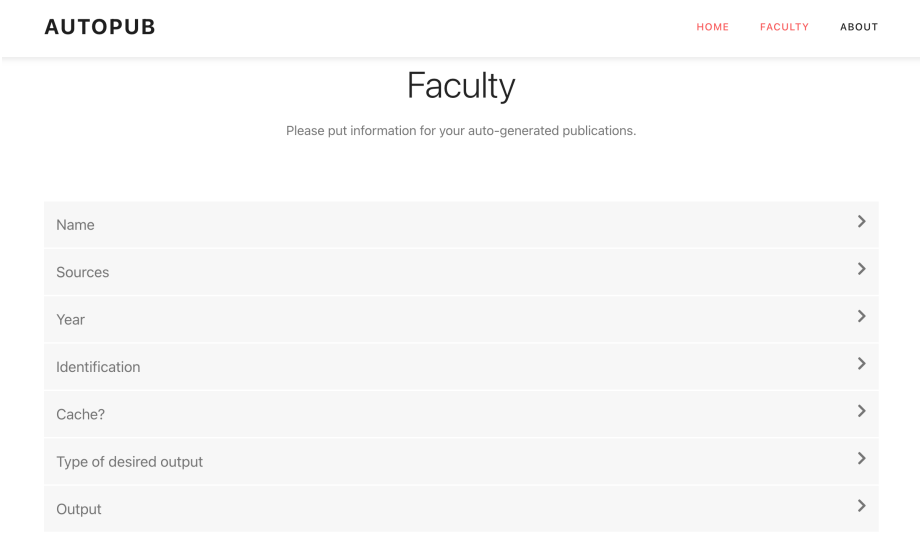


Figure 13: Faculty page.

10.2 Admin

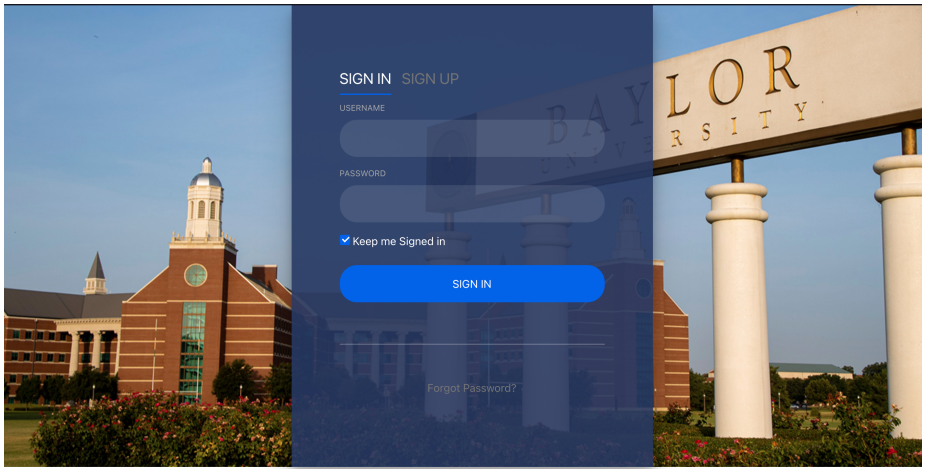


Figure 14: Login page.

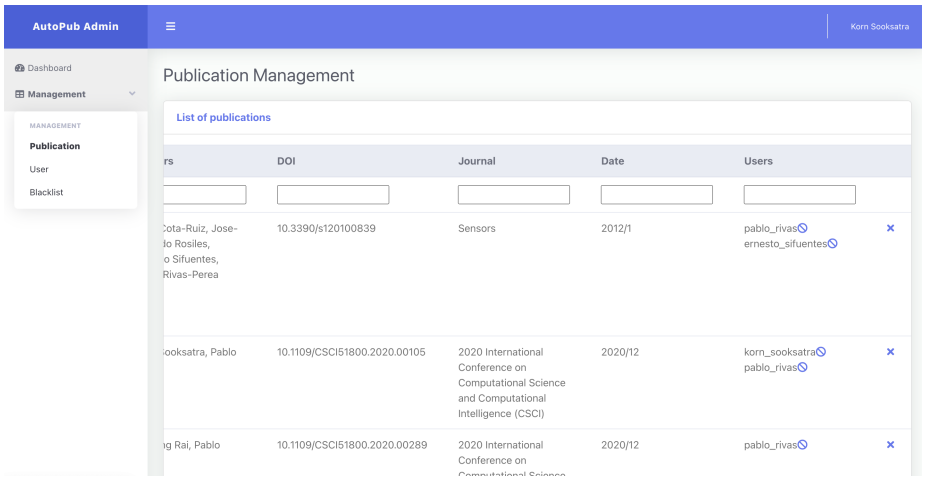


Figure 15: Publication page.

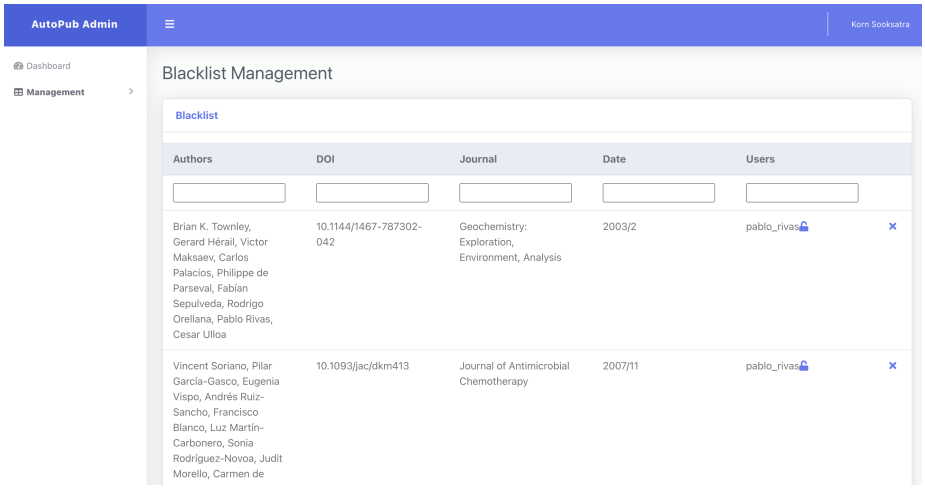


Figure 16: Blacklist page.

11 Backend

We are crawling the data from Google Scholar and Semantic Scholar. We are using htmlunit, jsoup to get the data from the web pages. Then, we store the data in the database. Figure 17 shows several rows from our dataabase. By getting one publication per minute we were able to get all papers without getting banned. The next step will be to create a RestAPI using Spring Boot and JPA.

	title [PK] character varying (750)	authors character varying (1000)	date	journal	volume	issue	pages	publisher	description character varying (5000)	citations	conference	book	url character varying (300)
5	A Comprehensive Enterpri...	Jan Svacina, Vincent Bushong, Dip...	2021	Inform...	739	[null]	245	Springer Nat...	One of the biggest challenges in code q...	[null]	[null]	[null]	https://scholar.googl...
6	A convolutional neural net...	Ryan Henning, Pablo Rivas-Perea, ...	2014/4/6	[null]	[null]	[null]	9-12	IEEE	We use Convolutional Neural Networks t...	11	2014 South...	[null]	https://scholar.googl...
7	A distributed localization ...	Juan Cota-Ruiz, J Rosiles, P Rivas, ...	2013	[null]	[null]	99	1-1	IEEE	We present a distributed localization alg...	48	[null]	[null]	https://scholar.googl...
8	A framework-based appro...	Jun Lin, Jonathan Drake, Hanil Kim...	2012/10/23	[null]	[null]	[null]	364-370	[null]	The interactive multimedia application h...	3	[null]	Proceed...	https://scholar.googl...
9	A Fuzzy Clustering Appro...	Mario I Chacon M, Pablo Rivas P, G...	2007/9/1	Engine...	15	1	[null]	[null]	This paper presents a new approach ai...	13	[null]	[null]	https://scholar.googl...
10	A low-complexity geometr...	Juan Cota-Ruiz, Jose-Gerardo Rosi...	2012/1	Sensors	12	1	839-862	Molecular Di...	This research presents a distributed and...	53	[null]	[null]	https://scholar.googl...
11	A meta-structure for supp...	Greg Speegle, Xiaojun Wang, Le Gr...	1998/7/6	[null]	[null]	[null]	89-102	Springer, Ber...	Multimedia databases include many typ...	10	British Natio...	[null]	https://scholar.googl...
12	A metamodeling approac...	Robert France, S Chosh, Eunjee So...	2003/9/15	IEEE s...	20	5	52-58	IEEE	Design patterns capture development so...	108	[null]	[null]	https://scholar.googl...
13	A nonlinear least squares ...	Pablo Rivas-Perea, Juan Cota-Ruiz,...	2014/8/1	Intern...	5	4	579-597	Springer Berl...	This paper studies the problem of hyper-...	11	[null]	[null]	https://scholar.googl...
14	A probabilistic model for ...	Pablo Rivas-Perea, Jose G Rosiles	2010/6/7	[null]	[null]	[null]	DMD4	Optical Soci...	We present a simple probabilistic model...	4	Digital Imag...	[null]	https://scholar.googl...

Figure 17: Database

12 Trello and Gantt Chart

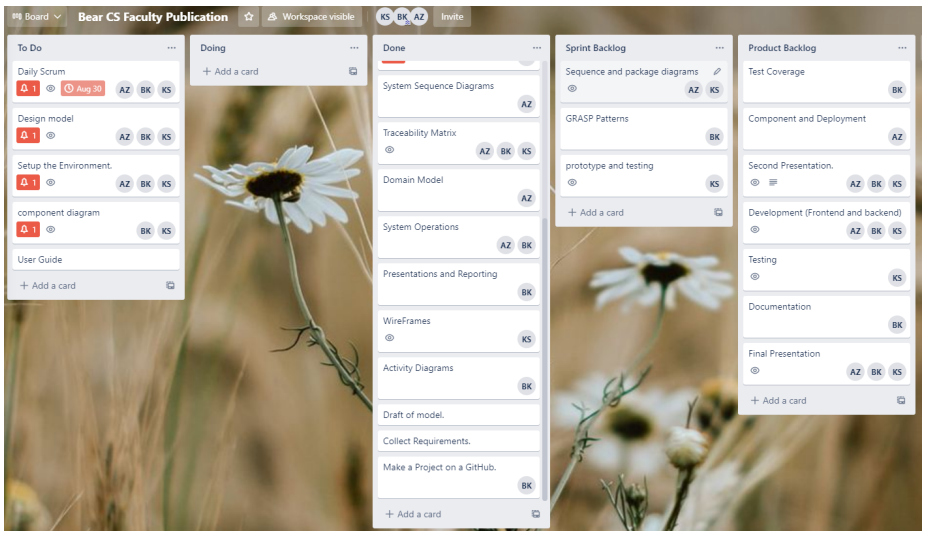


Figure 18: Trello page.

Untitled Gantt Project

Sep 28, 2021

http://

Project manager	
Project dates	Aug 25, 2021 - Nov 27, 2021
Completion	28%
Tasks	32
Resources	3

Tasks

Name	Begin date	End date
Analysis	8/25/21	9/29/21
Project Vision	8/25/21	8/26/21
Team Assembly	8/27/21	8/27/21
Infrastructure Initialization	8/30/21	9/1/21
Requirements analysis	8/27/21	9/6/21
Use Cases	9/2/21	9/10/21
Traceability Matrix	9/6/21	9/9/21
System Sequence Diagrams	9/9/21	9/13/21
System Operations	9/14/21	9/17/21
Wireframes	9/20/21	9/22/21
Domain Model	9/14/21	9/27/21
Activity Diagram	9/27/21	9/28/21
Presentation and Reporting	9/28/21	9/29/21
Design	10/4/21	10/29/21
Design Model	10/4/21	10/5/21
Sequence Diagrams	10/6/21	10/7/21
Package Diagrams	10/11/21	10/13/21
GRASP patterns	10/14/21	10/19/21
Test Coverage	10/20/21	10/22/21
Prototyping and testing	10/18/21	10/29/21
Component and Deployment	10/4/21	10/29/21
Presentation and Reporting	10/26/21	10/28/21
Implementation	10/29/21	11/26/21
Backend	11/1/21	11/19/21
User Interface	11/1/21	11/19/21
User Input validation	11/22/21	11/25/21
Imports/exports	11/17/21	11/18/21

Tasks

Name	Begin date	End date
Unit testing	11/22/21	11/22/21
Real Data Testing	11/23/21	11/24/21
System Test	11/22/21	11/24/21
Documentation	10/29/21	11/25/21
Presentation and Reporting	11/25/21	11/26/21

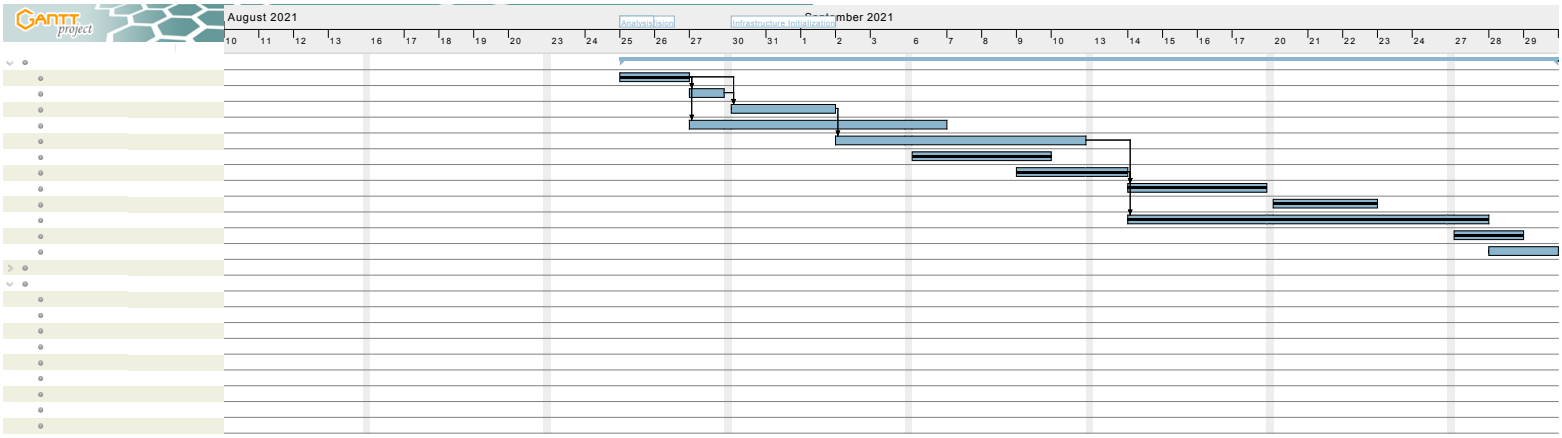
Resources

Name	Default role
Bikram Khanal	Team Lead
Alibek Zhakubayev	developer
Korn Sooksatra	Analyst

Untitled Gantt Project

Sep 28, 2021

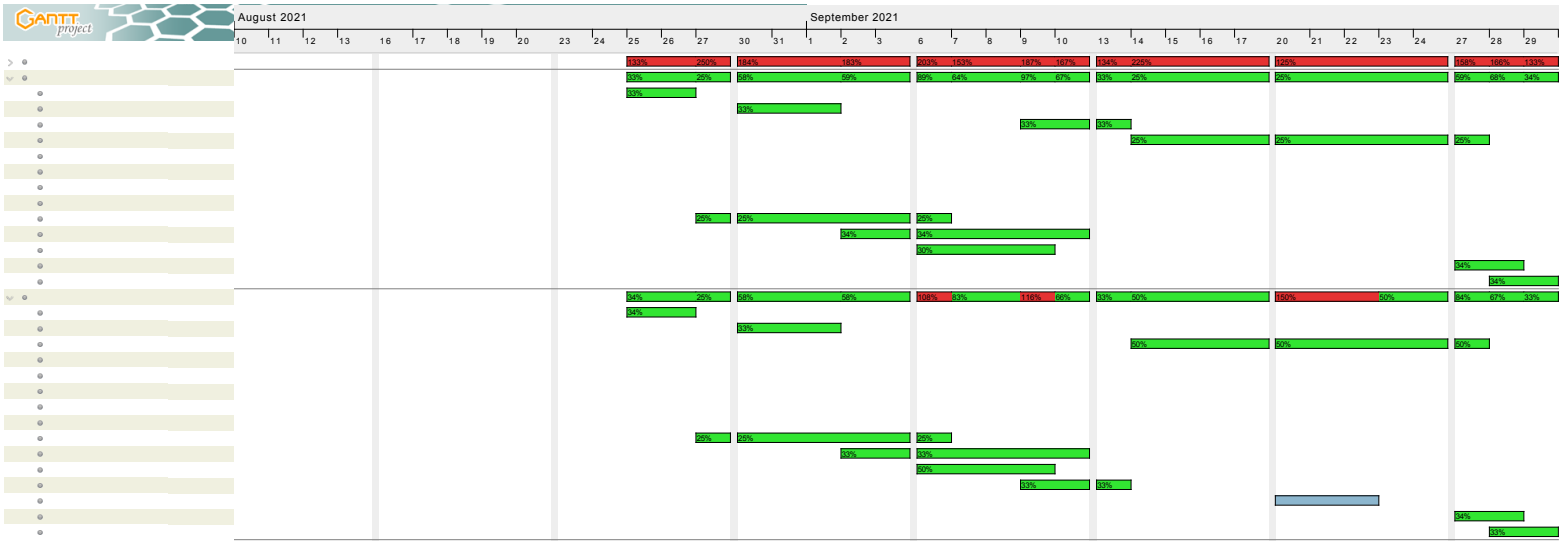
Gantt Chart



Untitled Gantt Project

Sep 28, 2021

Resources Chart



13 Artifacts

- **Frontend.** https://github.com/kornsook/AutoPub_Frontend
- **Other artifacts.** https://github.com/jeonbik/Software-Project.git

14 Work Hours

Team Members	Worked Hours
Bikram Khanal	10 Hours
Korn Sooksatra	11 Hours
Alibek Zhakubayev	11 Hours
Total	32 Hours

Table 3: Working Hours

Iteration 2

November 2021

1 Sequence Diagram

1.1 Get URL

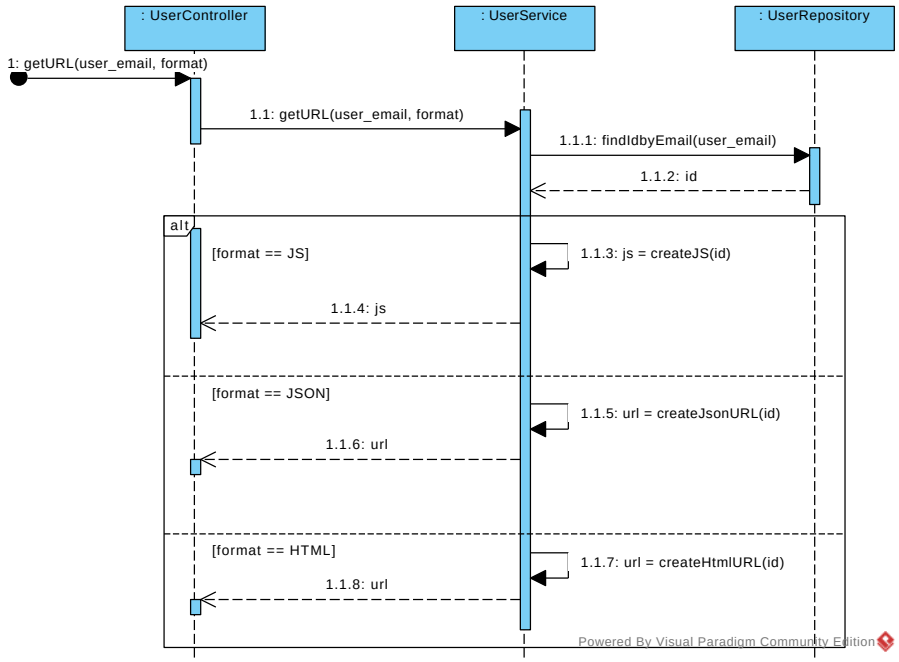


Figure 1: Sequence diagram of getting a URL

1.2 Request for Editing List

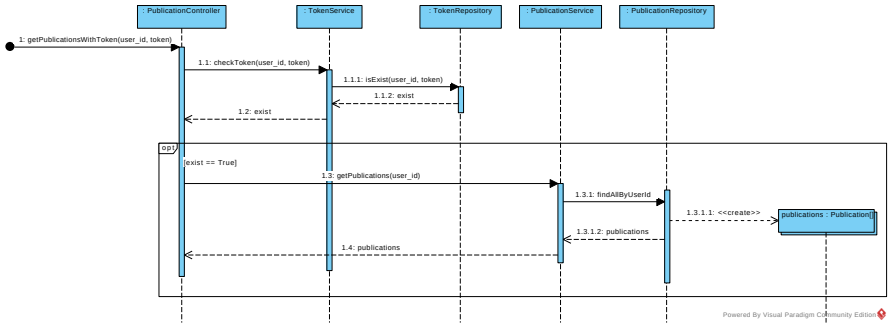


Figure 2: Sequence diagram of requesting for editing publication list

1.3 Get Publication List with Token

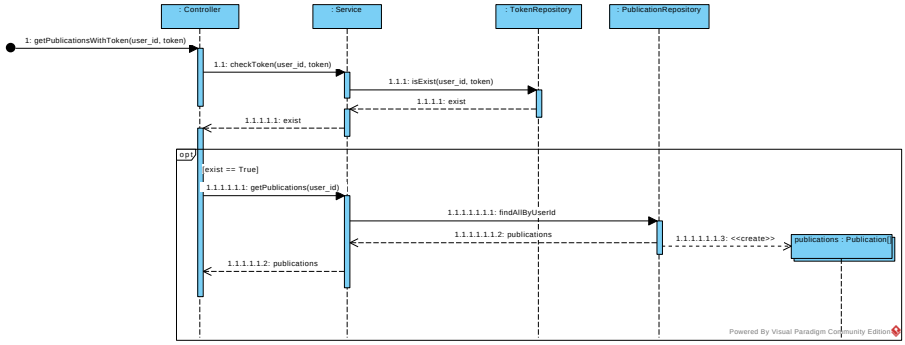


Figure 3: Sequence diagram of obtaining publication list with token

1.4 Read Abstract of Publication

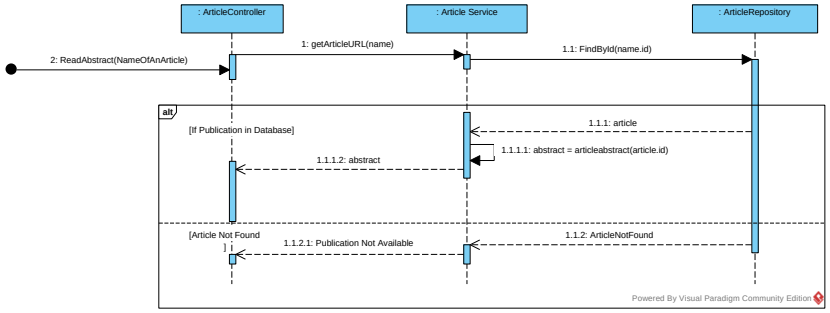


Figure 4: Sequence diagram of reading an abstract of a paper

1.5 Download Publication

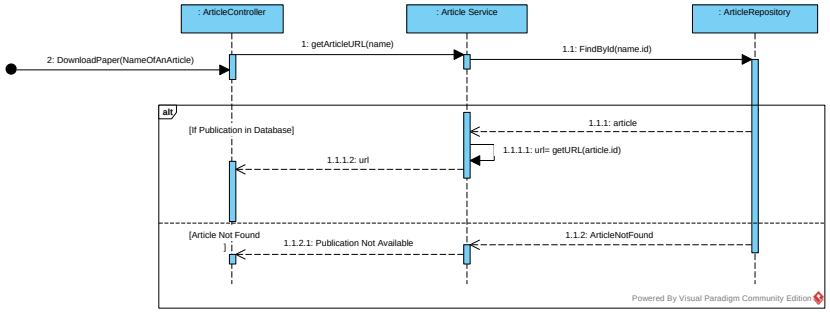


Figure 5: Sequence diagram of downloading a paper

1.6 Add item to do not publish list

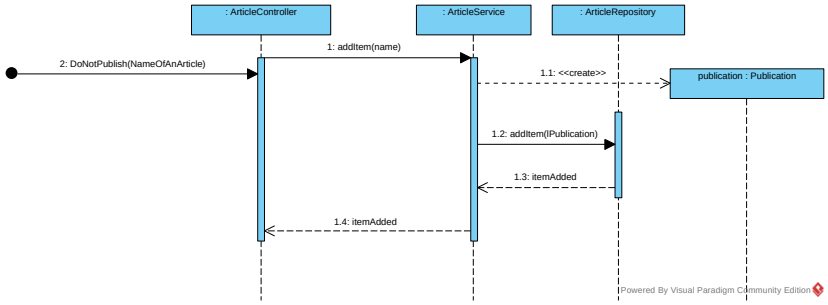


Figure 6: Sequence diagram of adding an item to do not publish list

1.7 Remove item to do not publish list

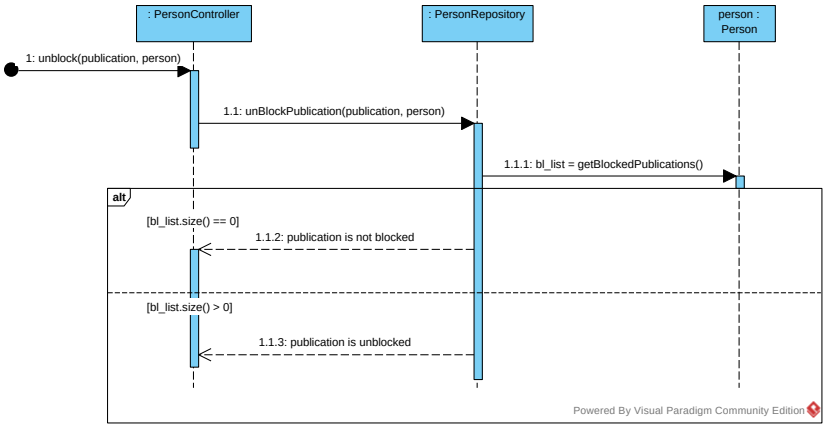


Figure 7: Sequence diagram of removing an item from publish list

1.8 Crawl publications for person

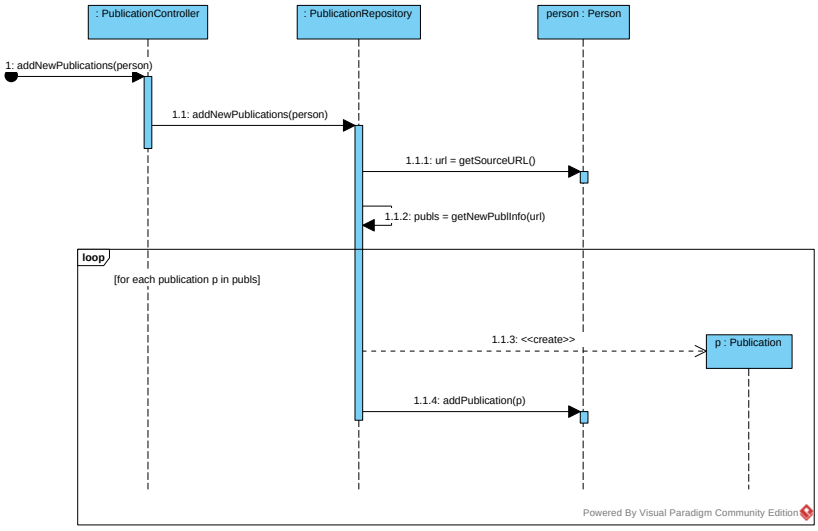


Figure 8: Sequence diagram of adding new publications to the user

1.9 Get BibTEX citation for publication

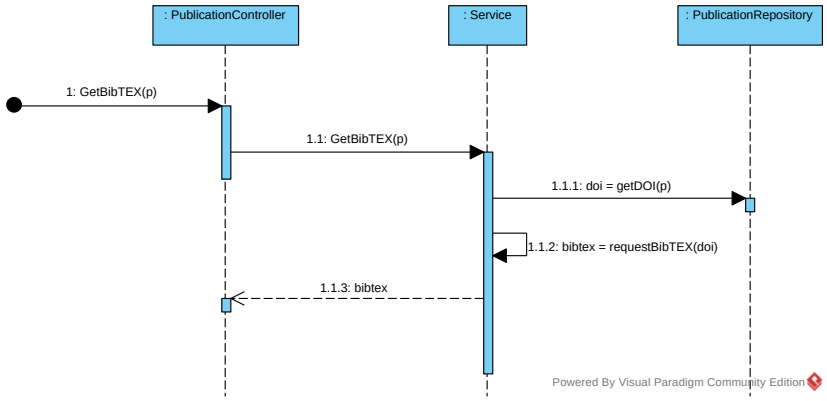


Figure 9: Sequence diagram of getting a publication BibTEX

2 ER Diagram

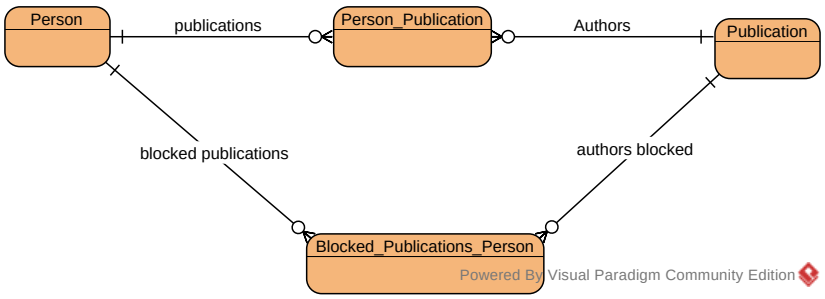


Figure 10: Design model

3 Architecture

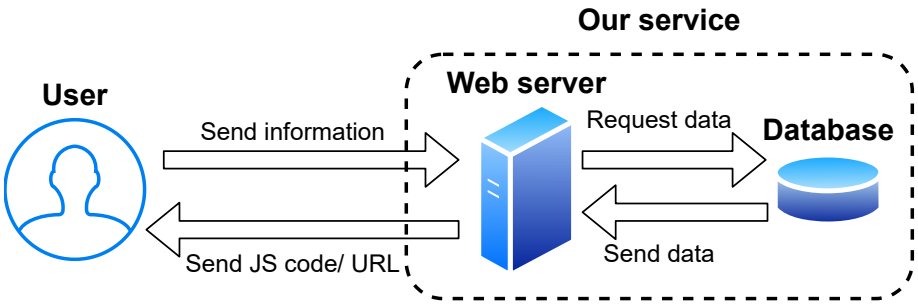


Figure 11: Architecture

4 Component Diagram

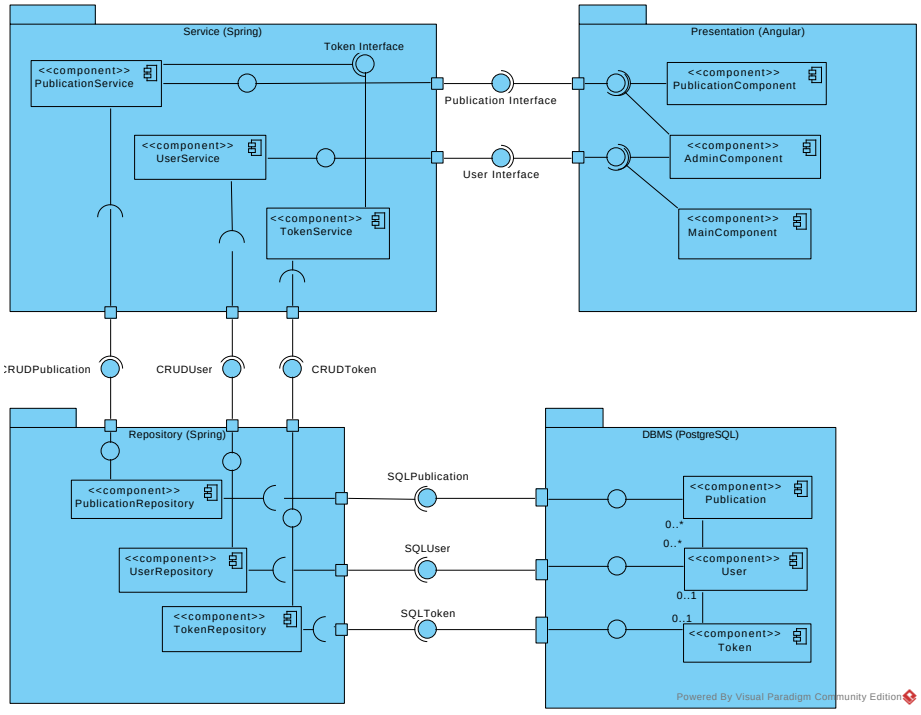


Figure 12: Component diagram

5 Object Constraint Language (OCL)

5.1 A token's expired date is always less than the current date

context: Token
invariant: expired_date < getTime()

5.2 A user can only block publications belonging to him/her

context: User
invariant: blacklist− >forAll(b|b.users− >includes(self))

5.3 Every publication has a title

context: Publication
invariant: title <> null

5.4 Every user has an email

context: User
invariant: email <> null

5.5 A user can own at most one token

context: User
invariant: token− >size() <= 1

6 GRASP Patterns

6.1 Controllers

- PersonController and PublicationController

6.2 Pure Fabrication

- Create PersonRepository, PublicationRepository and Services

6.3 Information Expert

- PersonRepository is an expert in all information regarding Person.
- PublicationRepository is an expert in all information regarding Publication.
- Service is an expert in all repositories.

6.4 Indirection

- Controller indirectly request data from repositories through the service.
- Service indirectly request data from DBMS through repositories.

7 Design model

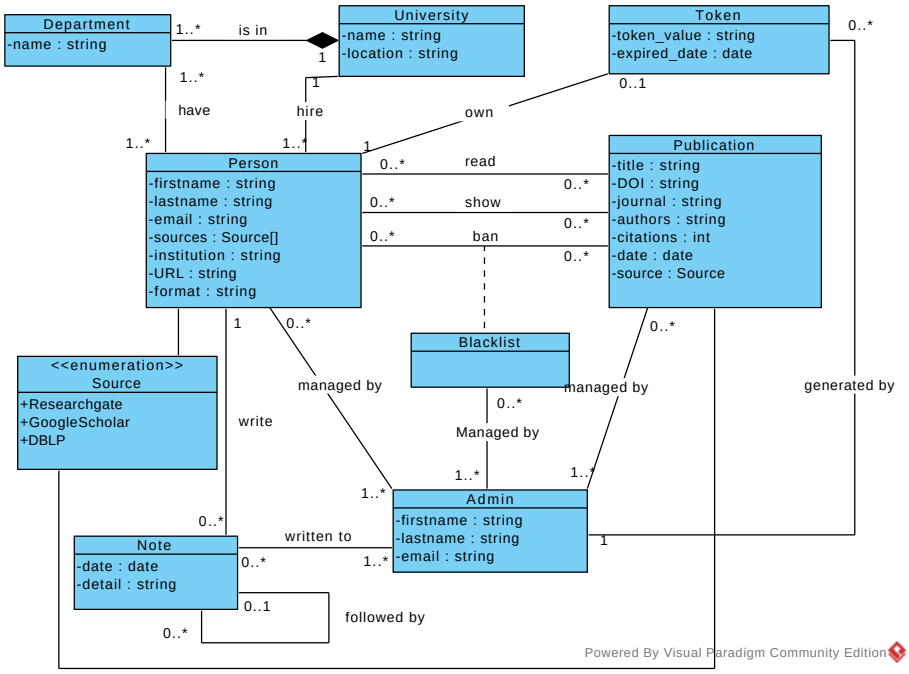


Figure 13: Design model

8 Gantt Chart and Trello

8.1 Trello

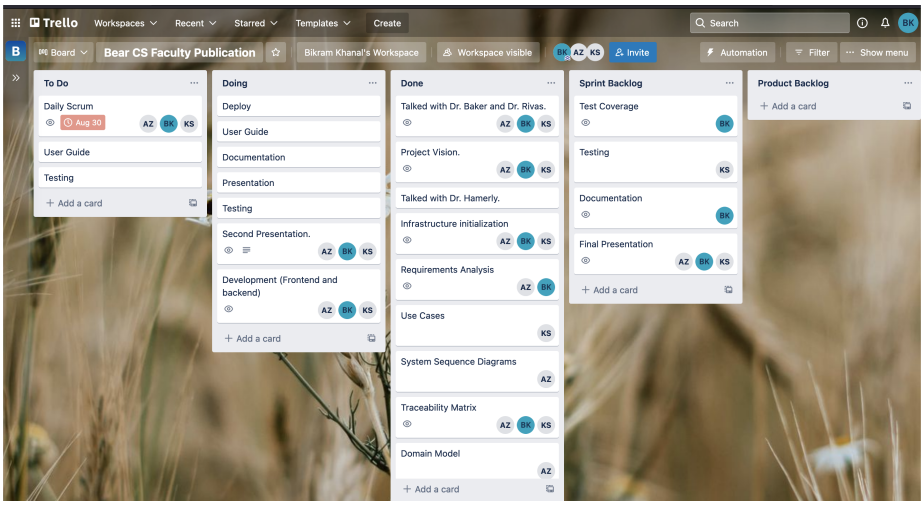


Figure 14: trello page.

8.2 Gantt Chart

Untitled Gantt Project

Nov 8, 2021

http://

Project manager	
Project dates	Aug 25, 2021 - Nov 27, 2021
Completion	78%
Tasks	32
Resources	3

Tasks

Name	Begin date	End date
Analysis	8/25/21	9/29/21
Project Vision	8/25/21	8/26/21
Team Assembly	8/27/21	8/27/21
Infrastructure Initialization	8/30/21	9/1/21
Requirements analysis	8/27/21	9/6/21
Use Cases	9/2/21	9/10/21
Traceability Matrix	9/6/21	9/9/21
System Sequence Diagrams	9/9/21	9/13/21
System Operations	9/14/21	9/17/21
Wireframes	9/20/21	9/22/21
Domain Model	9/14/21	9/27/21
Activity Diagram	9/27/21	9/28/21
Presentation and Reporting	9/28/21	9/29/21
Design	10/4/21	10/29/21
Design Model	10/4/21	10/5/21
Sequence Diagrams	10/6/21	10/7/21
Package Diagrams	10/11/21	10/13/21
GRASP patterns	10/14/21	10/19/21
Test Coverage	10/20/21	10/22/21
Prototyping and testing	10/18/21	10/29/21
Component and Deployment	10/4/21	10/29/21
Presentation and Reporting	10/26/21	10/28/21
Implementation	10/29/21	11/26/21
Backend	11/1/21	11/19/21
User Interface	11/1/21	11/19/21
User Input validation	11/22/21	11/25/21
Imports/exports	11/17/21	11/18/21

Tasks

Name	Begin date	End date
Unit testing	11/22/21	11/22/21
Real Data Testing	11/23/21	11/24/21
System Test	11/22/21	11/24/21
Documentation	10/29/21	11/25/21
Presentation and Reporting	11/25/21	11/26/21

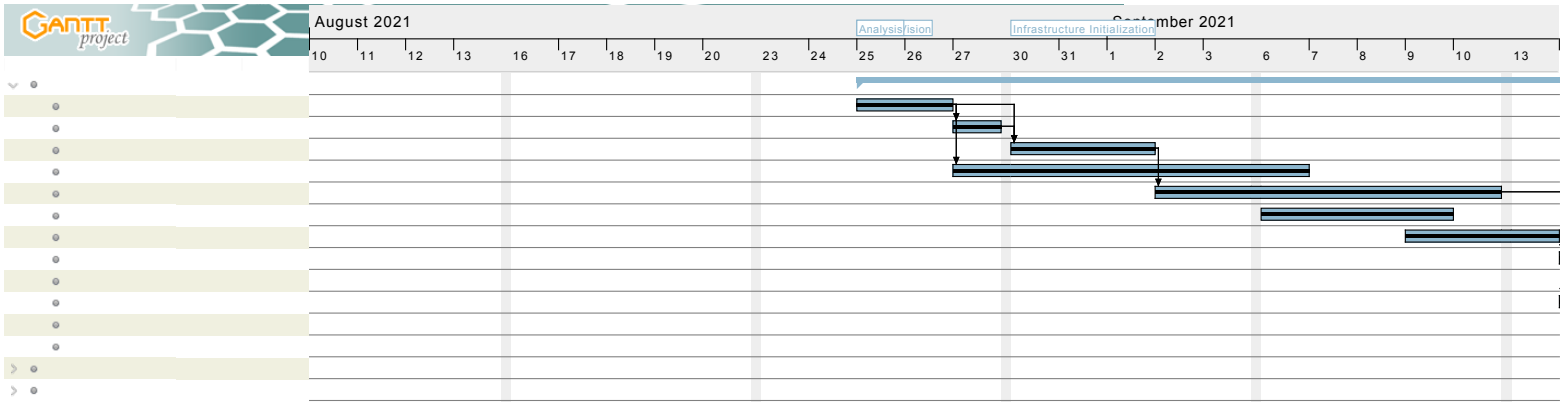
Resources

Name	Default role
Bikram Khanal	Team Lead
Alibek Zhakubayev	developer
Korn Sooksatra	Analyst

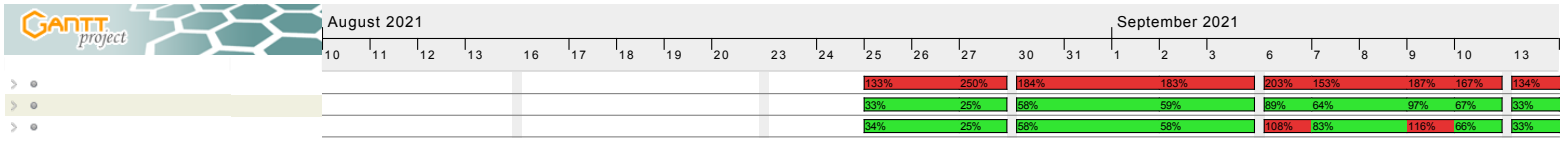
Untitled Gantt Project

Nov 8, 2021

Gantt Chart



Resources Chart



9 Demo

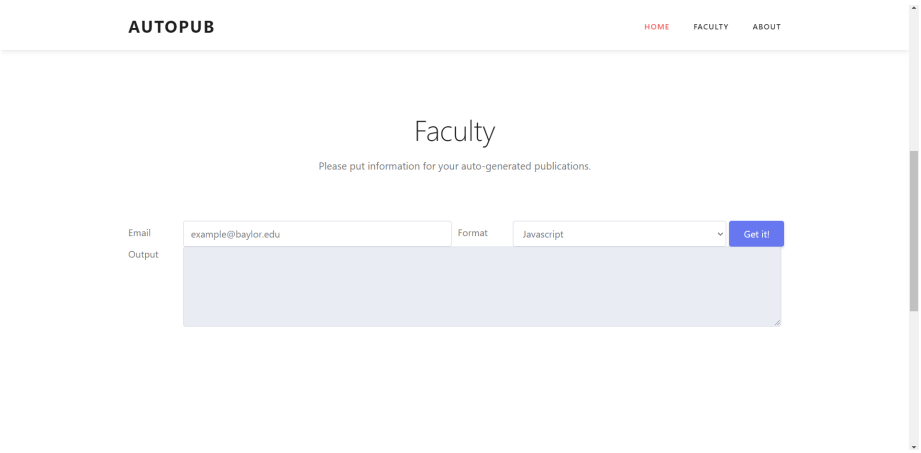


Figure 15: Front page

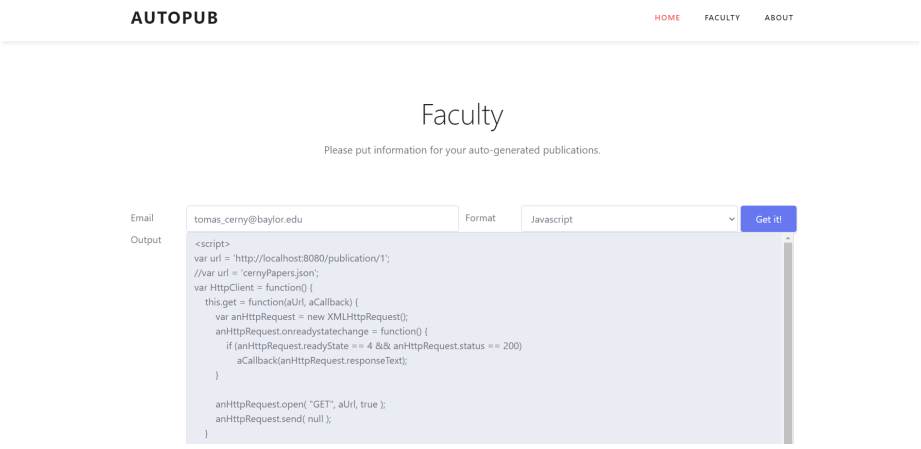


Figure 16: Generated javescript code

- Mary Lauren Benton, A. Abraham, A. Labella, P. Abbot, A. Rokas, J. Capra, "The influence of evolutionary history on human health and disease", Springer Science and Business Media LLC Nature reviews. Genetics.5 (2021): 269-283
- Kelsey S. Johnson, S. Hussein, Priyanka Chakraborty, Arvind Murugaanantham, Sheridan Mikhail, Giovanny Gonzalez, Shu-Sheng Song, M. Jolly, M. Toneff, Mary Lauren Benton, Yin C Lin, J. Taube, "Epithelial-mesenchymal plasticity through loss of CTCF motif accessibility and protein expression", Cold Spring Harbor Laboratory (2021)
- Keila Velázquez-Arcelay, Mary Lauren Benton, J. Capra, "Diverse functions associate with trans-species polymorphisms in humans", Cold Spring Harbor Laboratory (2021)
- Alexander Munden, Mary Lauren Benton, J. Capra, J. Nordman, "R-loop mapping and characterization during Drosophila embryogenesis reveals developmental plasticity in R-loop signatures", Cold Spring Harbor Laboratory (2021)
- S. P. Mcgrath, Mary Lauren Benton, Maryam Tavakoli, N. Tatonetti, "Predictions, Pivots, and a Pandemic: a Review of 2020's Top Translational Bioinformatics Publications", Georg Thieme Verlag KG Yearbook of medical informatics.01 (2021): 219-225
- Keila Velázquez-Arcelay, Mary Lauren Benton, J. Capra, "Diverse Functions Associate With Non-Coding Trans-species Polymorphisms in Humans", Research Square Platform LLC (2021)
- Lide Han, Xuefang Zhao, Mary Lauren Benton, Thaneer Perumal, Ryan L. Collins, G. Hoffman, Jessica S. Johnson, L. Sloofman, Harold Z. Wang, M. Stone, Schahram Jaroslav Michael Kristen J. Leaune Andrew Joseph D Akbarian Bendl Breen Brennand Brown Browne Buxbaum, S. Akbarian, J. Bendl, M. Breen, K. Brennand, Leanne Brown, A. Browne, J. Buxbaum, A. Charney, A. Chess, Lizette Couto, G. Crawford, Olivia Devillers, B. Devlin, Amanda Dobbyn, E. Domenici, M. Filosi, E. Flatow, N. Francoeur, J. Fullard, S. Gil, K. Girdhar, A. Gulyás-Kovács, R. Gur, C. Hahn, V. Haroutunian, M. Hauberg, L. Huckins, Rvky Jacobov, Yan Jiang, Bibi S. Kassim, Yungil Kim, L. Klei, R. Kramer, Mario Lauria, T. Lehner, D. Lewis, B. Lipska, Kelsey S. Montgomery, Royce B Park, C. Rosenblum, "Functional annotation of rare structural variation in the human brain", Springer Science and Business Media LLC Nature Communications.1 (2020)
- Lide Han, Xuefang Zhao, Mary Lauren Benton, Thaneer Perumal, Ryan L. Collins, G. Hoffman, Jessica S. Johnson, L. Sloofman, Harold Z. Wang, K. Brennand, H. Brand, S. Sieberts, S. Marenco, M. Peters, B. Lipska, P. Roussos, J. Capra, M. Talkowski, D. Ruderfer, "Functional annotation of rare structural variation in the human brain", Cold Spring Harbor Laboratory (2019)
- Mary Lauren Benton, Sai Charan Talipineni, D. Kostka, J. Capra, "Genome-wide enhancer annotations differ significantly in genomic distribution, evolution, and function", Springer Science and Business Media LLC BMC genomics.1 (2019)
- Lide Han, Xuefang Zhao, Mary Lauren Benton, Thaneer Perumal, Ryan L. Collins, H. Brand, G. Hoffman, Jessica S. Johnson, L. Sloofman, K. Brennand, S. Sieberts, S. Marenco, M. Peters, B. Lipska, P. Roussos, J. Capra, M. Talkowski, D. Ruderfer, "25 FUNCTIONAL ANNOTATION OF RARE STRUCTURAL VARIATION IN THE HUMAN BRAIN", Elsevier BV European Neuropsychopharmacology (2019): S72-S73
- Mary Lauren Benton, Sai Charan Talipineni, D. Kostka, J. Capra, "Genome-wide Enhancer Maps Differ Significantly in Genomic Distribution, Evolution, and Function", Cold Spring Harbor Laboratory (2017)
- Mary Lauren Benton, J. Capra, E. Hodges, J. Hughey, "Genome-wide Enhancer Maps Differ Significantly in their Genomic Distribution, Evolution, and Function By"
- Andrea A. Perreault, Mary Lauren Benton, M. Koury, S. Brandt, Bryan J Venters, "Epo reprograms the epigenome of erythroid cells.", Elsevier BV Experimental hematology (2017): 47-62

Figure 17: List of publications

10 Github

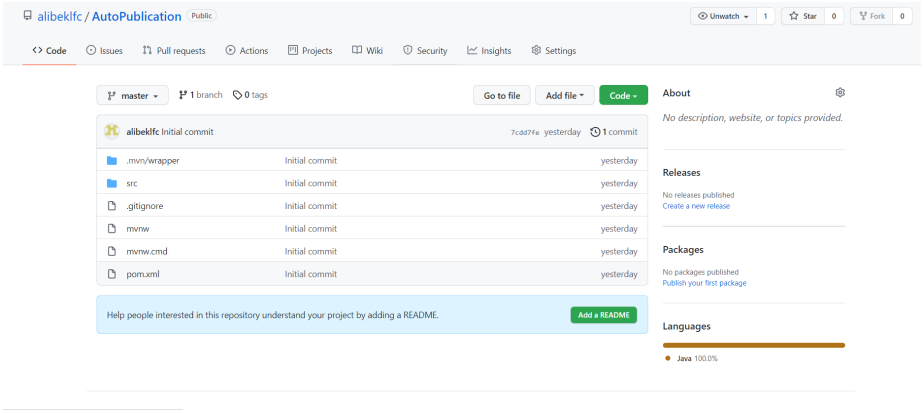


Figure 18: Github page of the backend

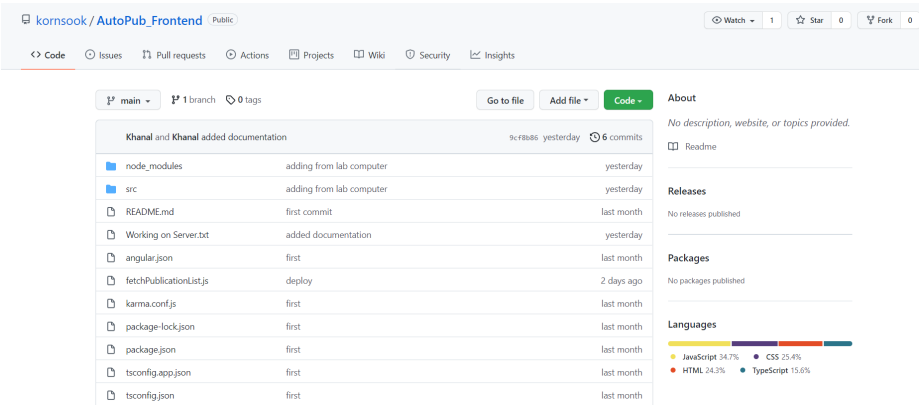


Figure 19: Github page of the frontend

11 Work Hours

Team Members	Worked Hours
Bikram Khanal	20 Hours
Korn Sooksatra	20 Hours
Alibek Zhakubayev	20 Hours
Total	60 Hours

Table 1: Working Hours