**University of Louisville**

**Iteration 5**

UofL Research & Innovation Website Renovation Vision (Small Project)

Team Added Value

Andy Lin, Celeste Menard, Baian Khanjar, Sohal Patel, Della Wang, and Emily Wantland

CIS 310-01

**University of Louisville**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 27/Jan/20 | 1.0 | First Iteration | Ruomei Wang |
| 10/Feb/20 | 2.0 | Second Iteration | Sohal Patel |
| 1/Mar/20 | 3.0 | Inception Spec | Ruomei Wang |
| 20/Mar/20 | 4.0 | Third Iteration | Emily Wantland |
| 8/April/20 | 5.0 | Fifth Iteration | Emily Wantland |

**Table of Contents**

1. Class Diagram 3-19
2. Database Design and Data Definitions 19-31
3. User Interface Navigation Diagram and Screen Layouts 32-49
4. Gantt Chart 50
5. Prototype 51
6. **Class Diagram**
   * The class diagram includes all probable domain classes. It displays the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects. A class is a description of a group of objects all with similar roles in the system. The diagram is further explained with CRC Cards which are a brainstorming tool.
   * The class diagram outlines the relationships and cardinality of all the classes that play an important role in the functionality of the University of Louisville Research & Innovation website.

* For example, the Industry Proposal class is associated with the Partnership class. When the industry proposal is created, the unique “industry email” attribute will be stored in the Partnership class as well.
* Each diagram includes the attributes the function requires and the methods to realize the function.
* For example, the Industry Proposal class contains the attributes that the Industry user needs to provide when he/she submit the industry proposal, as well as the methods he/she can utilize when they are filling out and submitting their proposal.



**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Student Research Application | **ID**:1 | | **Type**:Concrete, Domain |
| **Description**: Students will be able to apply for research opportunities with researchers. | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Upload()  Submit()  Edit()  Delete() | | **Collaboration**  Research | |

**Back**

|  |
| --- |
| **Attributes:**  Research application ID  Student ID  Research ID  Research description |
| **Relationships:**  **Generalization (a-kind-of):**  Form  **Aggregation (has-parts):** N/A  **Other Associations:** Research |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Research Application | **ID**: 2 | | **Type:**  Concrete, Domain |
| **Description:**  The research table includes a research ID which will be unique for each research. Then researcher ID to identify the student who is participating in the research. Then information about the research starting with research name and area. Then the cost of this research and lastly the department that this research is part of. | | **Associated Use Cases:** 3 | |
| **Responsibilities**  Enter the application information  Edit the application  Submit the application  Delete the application | | **Collaboration**  Researcher  Partnership  Student research application | |

**Back**

|  |
| --- |
| **Attributes:**  Research ID  Researcher ID  Research name  Research area  Research cost  Department |
| **Relationships:**  **Generalization (a-kind-of):**  Form  **Aggregation (has-parts):** N/A  **Other Associations:** Researcher  Partnership  Student research application |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Student | **ID:** 3 | | **Type:**  Concrete, Domain |
| **Description:**  Student table contains the student information so we have student information in our databases. It contains first of all Student ID which is the unique Id for each student. The student roll number is also unique for each student. Student table must also contain student name, address and contact information. This is important to send mails that contain important information to the student. Student email is also the most important thing that the student will receive important emails about their application. It also will contain a student class and department. | | **Associated Use Cases:** 2 | |
| **Responsibilities**  Create grant student application | | **Collaboration**  Student Log In  Student Grant Application | |

**Back**

|  |
| --- |
| **Attributes:**  Student ID  Student roll number  Student name  Student address  Student contact  Student email  Student class  Student department |
| **Relationships:**  **Generalization (a-kind-of):**  Connection  **Aggregation (has-parts):** N/A  **Other Associations:** Student Log In  Student Grant Application |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Student Login | **ID:** 4 | | **Type:**  Concrete, Domain |
| **Description:**  The login table contains User ID-int(7) as a unique Id for each student and Password- int(20) attributes. The User class contains two methods, enter and submit. It interacts with the User Data class to login, and the User Data class interacts with the Roles class to distribute permissions. | | **Associated Use Cases:**  1 | |
| **Responsibilities:**  Enter()  Submit()  Cancel() | | **Collaboration:**  Admin | |

**Back**

|  |
| --- |
| **Attributes:**  Admin ID  Admin password  Admin role |
| **Relationships:**  **Generalization (a-kind-of):**  Form  **Aggregation (has-parts):** N/A  **Other Associations:** Admin |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Data | **ID:** 5 | | **Type:** Concrete, Domain |
| **Description:**  The admin can log into their account and backup, restore, or rebuild the website in the case of data loss | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Backup data  Restore data | | **Collaboration**  Admin | |

**Back**

|  |
| --- |
| **Attributes:**  Data entry ID  Calendar data  Data entry creation date |
| **Relationships:**  **Generalization (a-kind-of):** Data  **Aggregation (has-parts):** N/A  **Other Associations:** Admin |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Researcher Funding Application | **ID:** 6 | | **Type:** Concrete, Domain |
| **Description:**  Researchers can apply for funding to fuel their research | | **Associated Use Cases:** 3 | |
| **Responsibilities**  Create funding application  Submit funding application  Edit funding application  Re-submit funding application  Delete funding application | | **Collaboration**  Research  Researcher  Grant | |

**Back**

|  |
| --- |
| **Attributes:**  Researcher application ID  Research ID  Researcher ID  Department  Research name  Participating Student ID  Requesting support  Receiving support  Research cost  Funding notes  Application date |
| **Relationships:**  **Generalization (a-kind-of):** Form  **Aggregation (has-parts):** N/A  **Other Associations:** Research, Researcher, Grant |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Grant | **ID:** 7 | | **Type:** Concrete, Domain |
| **Description:**  The Grant class stores the information of grants | | **Associated Use Cases:** 3 | |
| **Responsibilities**  Add grants  Edit grants  Delete grants | | **Collaboration**  Admin  Researcher Funding Application  Student Grant Application | |

**Back**

|  |
| --- |
| **Attributes:**  Grant ID  Admin ID  Grant name  Grant type  Grant amount  Grant period  Grant authority  Grant description |
| **Relationships:**  **Generalization (a-kind-of):** Document  **Aggregation (has-parts):** N/A  **Other Associations:** Admin, Researcher Funding Application, Student Grant Application |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Research | **ID:** 8 | | **Type:** Concrete, Domain |
| **Description:**  The Research class stores the allocation of researches and researchers | | **Associated Use Cases:** 4 | |
| **Responsibilities**  Enter research/researcher information  Submit information | | **Collaboration**  Researcher  Partnership  Student Research Application  Researcher Funding Application | |

**Back**

|  |
| --- |
| **Attributes:**  Research ID  Researcher ID  Research name  Research area  Research cost  Department |
| **Relationships:**  **Generalization (a-kind-of):** Connection  **Aggregation (has-parts):** N/A  **Other Associations:** Researcher, Partnership, Student Research Application, Researcher Funding Application |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Partnership | **ID:** 9 | | **Type:** Concrete, Domain |
| **Description:**  The partnership is generated after the industry proposal is accepted | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Review industry proposal  Assign industry proposal to researcher | | **Collaboration**  Research | |

**Back**

|  |
| --- |
| **Attributes:**  Partnership ID  Partnership name  Industry email  Research ID  Partnership description |
| **Relationships:**  **Generalization (a-kind-of):** Connection  **Aggregation (has-parts):** N/A  **Other Associations:** Research |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Industry Proposal | **ID:** 10 | | **Type:** Concrete, Domain |
| **Description:**  Industry users can create/edit/delete proposals to initiate research | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Enter proposal information  Submit proposal  Edit proposal  Re-submit proposal  Delete proposal  Contact UofL staff | | **Collaboration**  Partnership | |

**Back**

|  |
| --- |
| **Attributes:**  Industry email  Industry representative first name  Industry representative last name  Company name  Company website  Industry representative phone number  Industry focused area  Industry proposal comments |
| **Relationships:**  **Generalization (a-kind-of):** Forms  **Aggregation (has-parts):** N/A  **Other Associations:** Partnership |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Researcher | **ID:** 11 | | **Type:** Concrete, Domain |
| **Description:**  The information of Researcher users | | **Associated Use Cases:** 3 | |
| **Responsibilities**  Enter Researcher user information  Submit Researcher user information | | **Collaboration**  Profile  Research  Researcher Funding Application | |

**Back**

|  |
| --- |
| **Attributes:**  Researcher ID  Researcher password  Researcher name  Researcher address  Researcher email  Researcher phone number  Researcher class  Researcher industry |
| **Relationships:**  **Generalization (a-kind-of):** Person  **Aggregation (has-parts):** N/A  **Other Associations:** Profile, Research, Researcher Funding Application |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Newsletter | **ID:** 12 | | **Type:** Concrete, Domain |
| **Description:**  The admin can create/edit/delete newsletters to showcase UofL research accomplishments | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Submit newsletters  Add newsletters  Edit newsletters  Delete newsletters | | **Collaboration**  Admin | |

**Back**

|  |
| --- |
| **Attributes:**  Newsletter ID  Newsletter title  Newsletter author  Newsletter description |
| **Relationships:**  **Generalization (a-kind-of):** Webpages  **Aggregation (has-parts):** N/A  **Other Associations:** Admin |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Profile | **ID:** 13 | | **Type:** Concrete, Domain |
| **Description:**  The researchers can create profile to showcase their works | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Enter the profile information  Edit the profile  Submit the profile | | **Collaboration**  Researcher | |

**Back**

|  |
| --- |
| **Attributes:**  Profile ID  Researcher ID  Researcher name  Researcher industry  Researcher major  Researcher information |
| **Relationships:**  **Generalization (a-kind-of):** Webpages  **Aggregation (has-parts):** N/A  **Other Associations:** Researcher |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Admin | **ID:** 14 | | **Type:** Concrete, Domain |
| **Description:**  The information of Admin users | | **Associated Use Cases:** 7 | |
| **Responsibilities**  Add grants  Modify grants  Delete grants  Submit newsletters  Add newsletters  Edit newsletters  Delete newsletters | | **Collaboration**  Newsletter  Grant  Research & Innovation Webpages  Event  Surveys  Data  Admin Login | |

**Back**

|  |
| --- |
| **Attributes:**  Admin ID  Grant ID  Event ID  Newsletter ID  Survey ID  Admin name  Admin password  Admin role |
| **Relationships:**  **Generalization (a-kind-of):** Person  **Aggregation (has-parts):** Admin Login  **Other Associations:** Newsletter, Grant, Research & Innovation Webpages, Event, Surveys |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Event | **ID:** 15 | | **Type:** Concrete, Domain |
| **Description:**  Admins can create/edit/delete the events | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Enter the event information  Add events  Edit events  Delete events  Submit events | | **Collaboration**  Admin | |

**Back**

|  |
| --- |
| **Attributes:**  Event ID  Event name  Event location  Event date  Event time  Event details |
| **Relationships:**  **Generalization (a-kind-of):** Events  **Aggregation (has-parts):** N/A  **Other Associations:** Admin |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Survey | **ID:** 16 | | **Type:** Concrete, Domain |
| **Description:**  Admins can create/edit/delete surveys that are designed for students to take | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Enter the survey information  Publish the survey  Edit the survey  View all surveys  Re-publish the survey  Delete the survey  Add a question  Add an option  Delete a question  Delete an option | | **Collaboration**  Admin | |

**Back**

|  |
| --- |
| **Attributes:**  Admin ID  Survey title  Survey area  Question  Option |
| **Relationships:**  **Generalization (a-kind-of):** Surveys  **Aggregation (has-parts):** N/A  **Other Associations:** Admin |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Admin Login | **ID:** 17 | | **Type:** Concrete, Domain |
| **Description:**  Admins can log in to the Research & Innovation website | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Enter username  Enter password  Submit login | | **Collaboration**  Admin | |

**Back**

|  |
| --- |
| **Attributes:**  Admin ID  Admin password  Admin role |
| **Relationships:**  **Generalization (a-kind-of):** Form  **Aggregation (has-parts):** N/A  **Other Associations:** Admin |

**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Research & Innovation Webpages | **ID:** 18 | | **Type:** Concrete, Domain |
| **Description:**  The webpages will be organized by the admins | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Add a link | | **Collaboration**  Admin  Webpage Link | |

**Back**

|  |
| --- |
| **Attributes:**  Webpage ID  Admin ID  Link ID  Event ID  Survey ID  Newsletter ID  Social media ID  Profile ID |
| **Relationships:**  **Generalization (a-kind-of):** webpages  **Aggregation (has-parts):** Webpage Links  **Other Associations:** Admin |

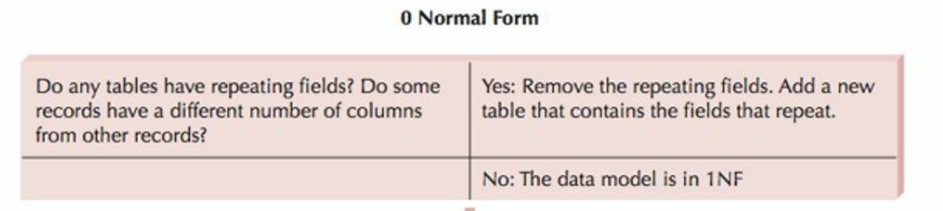
**Front**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class name:**  Webpage Links | **ID:** 19 | | **Type:** Concrete, Domain |
| **Description:**  The admin can add links to the webpages to different UofL departments or different content | | **Associated Use Cases:** 1 | |
| **Responsibilities**  Add a link | | **Collaboration**  Research & Innovation Webpages | |

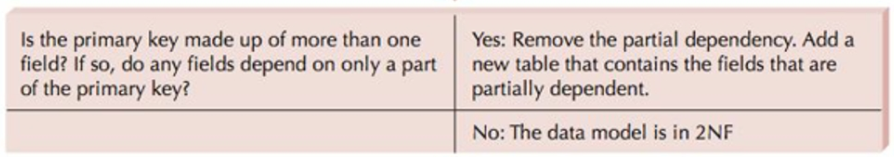
**Back**

|  |
| --- |
| **Attributes:**  Link ID  Webpage ID  Admin ID  Link page name  Link page department |
| **Relationships:**  **Generalization (a-kind-of):** webpages  **Aggregation (has-parts):** N/A  **Other Associations:** Research & Innovation Webpages |

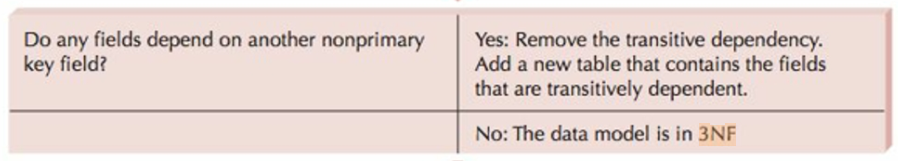
1. **Database Design and Database Definitions**
   * The Data Management Layer Design take part in different steps:



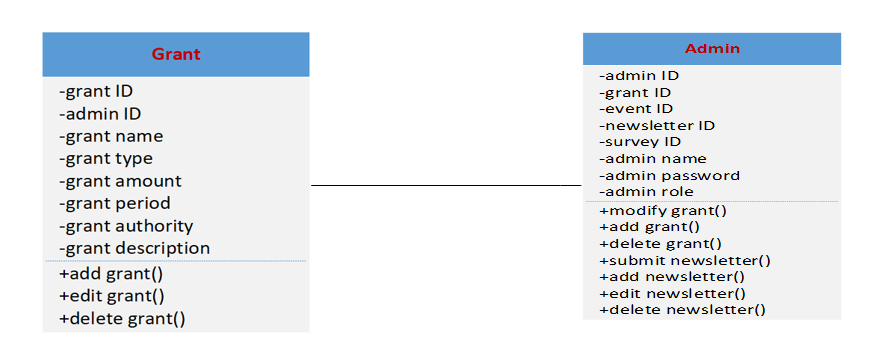




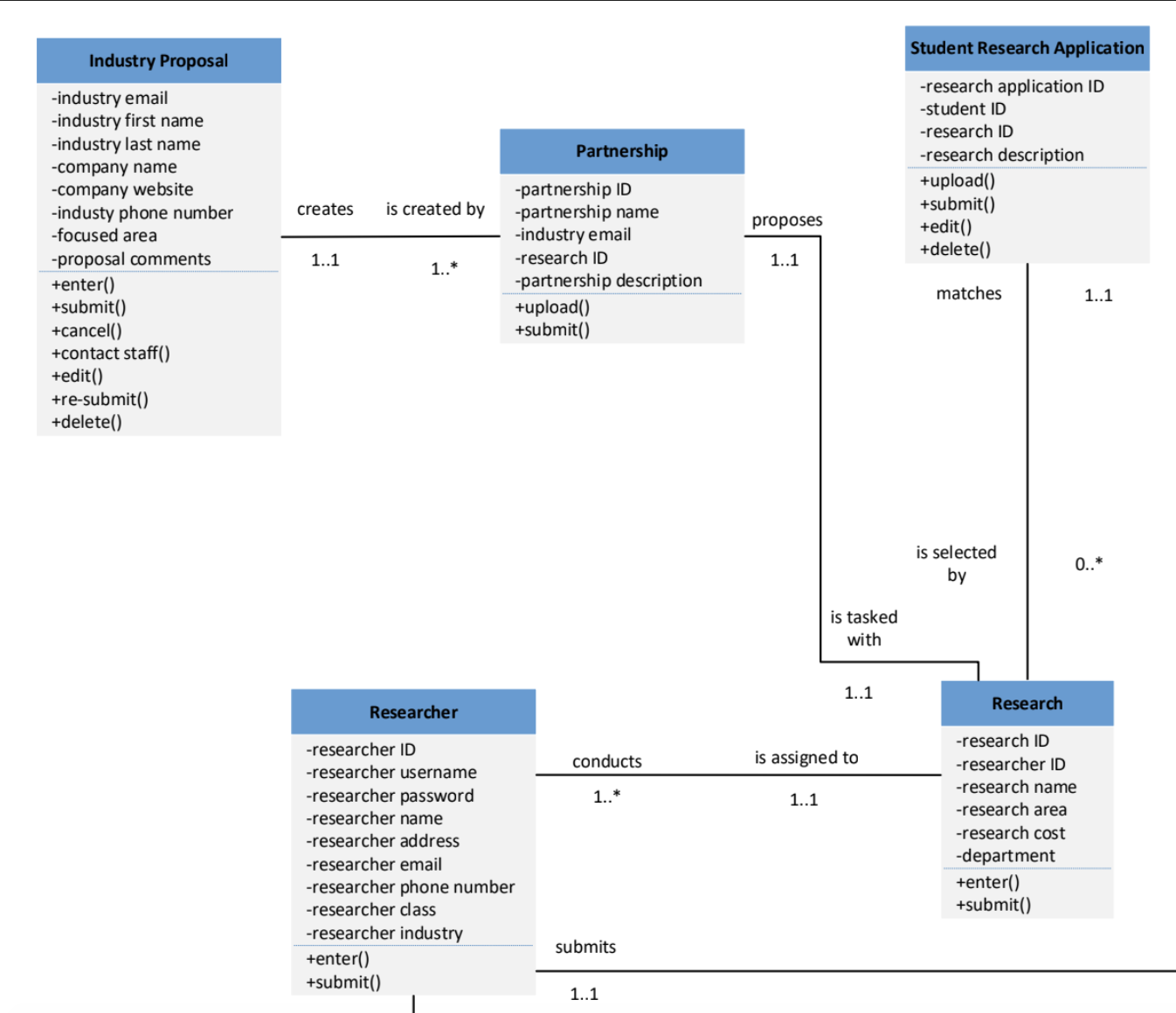






**Example**

In these two tables the admin ID in the admin table will serve as part of the primary key of the Grant table. Now if we have any records that are repeated will delete it. We did not have any repeats.



Research ID will serve as the primary key of the Researcher table.

Research ID will serve as part of the primary key of Research.

Research ID will serve as Foreign key of Research.

Researcher ID will serve as part of the primary key of Research.

Researcher ID will serve as Foreign key of Research.

Research ID will serve as part of the primary key of the Partnership.

Research Application ID will serve as part of the primary key of the Research application.

Research ID will serve as Foreign key of the Research application.

Research ID will serve as part of the primary key of the Partnership.

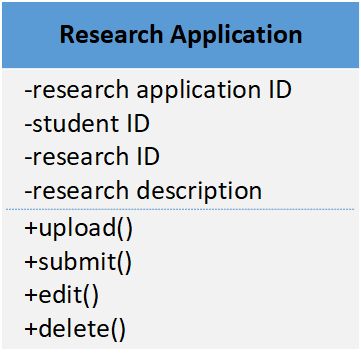
Partnership ID will serve as part of the primary key of the Partnership.

Partnership ID will serve as Foreign key of the Partnership.

**Eliminate Redundancy**

Researcher Username and password was moved to Researcher Table to eliminate redundancy. Research description was moved from research table to just research application table to eliminate redundancy.

**Data Descriptions**



**Research Application Table**

Student ID - Int(9)

Research ID- Int(9)

Research Description - Varchar(200)

**Narrative**

The research application table, we decided that we will have tables for each type of application since we have other applications like student application and funding application. This table is just to contain the student ID which will be int 9 numbers as it is always like this and we do not want to waste space in the memory. Also this table will contain a Research ID to identify each research that the university is doing. Each research will have a unique ID. We decided that it will also be 9 numbers. And lastly research description to add any notes about this research and it is char type 200.



**Researcher Funding Application Table**

Project Name - Varchar(30)

Department - Varchar(30)

Research Area - Varchar(30)

Researcher Name - Varchar(40)

Student ID - Int(9)

Requesting Support - int(8)

Receiving Support -int(8)

Research Cost - int(8)

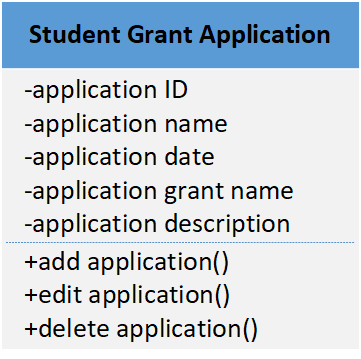
Notes - Varchar(200)

Date -Date(8)

**Narrative**

Funding application table contains the project name to define each project that they will fund. Department Varchar(30) to know this research is for which department in the university.

Research Area Varchar(30) to specify the research area exactly that this research will serve to. Researcher Name Varchar(40), Student ID Int(9). Requesting Support int(8) to specify the amount of the requesting support. Receiving Support int(8)if the researcher had support before. Research Cost int(8) to estimate how much this research will cost. Notes Varchar(200)we make sure that on almost every table here has a note which is to give the applicant an option to add anything that he/she wants. Date-Date(10)it is like a sign and to specify when the applicants complete their application.



**Student Grant Application Table**

Application ID - int(9)

Application Name - Varchar(10)

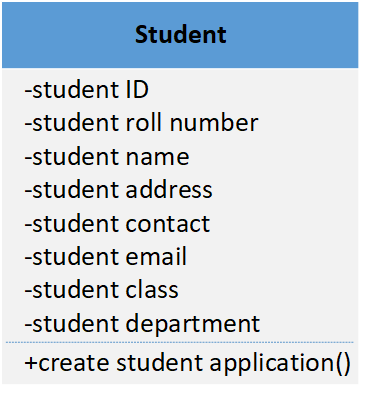
Application Date - Date(8)

Application Grant Name -Varchar(20)

Application Description -Varchar(200)

**Narrative**

The Student Application Table contains an application ID which is going to be unique for each application. Then application name and details about the application. Date to specify the date that the applicant completed and sent their application. Application details to let the applicant add whatever they want.



**Student Table**

student ID - int(9)

student roll number - int(6)

student name - Varchar(20)

student address - Varchar(30)

student contact - int(10)

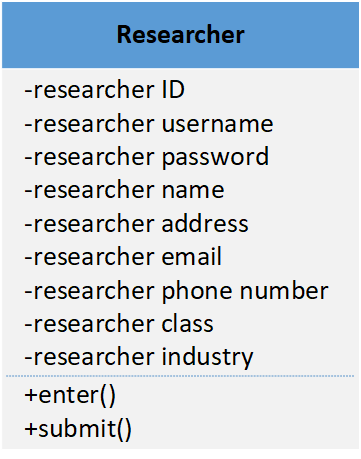
student email - varchar(20)

student class - varchar(10)

student department - varchar(20)

**Narrative**

Student table contains the student information so we have student information in our databases. It contains first of all Student ID which is the unique Id for each student. The student roll number is also unique for each student. Student tables must also contain student name, address and contact information. This is important to send mails that contain important information to the student. Student email is also the most important thing that the student will receive important emails about their application. It also will contain a student class and department.



**Researcher Table**

researcher ID - int(7)

researcher username - Varchar(6)

researcher password - Varchar(20)

researcher name - Varchar(15)

researcher address - Varchar(30)

researcher email - Varchar(20)

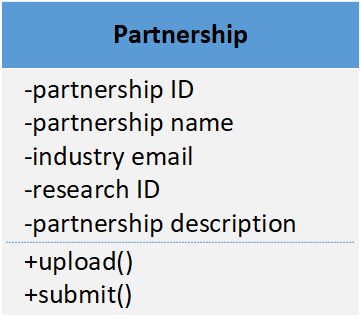
researcher phone number - int(10)

researcher class - Varchar(10)

researcher industry - Varchar(10)

**Narrative**

Researcher table will contain the main information about the researcher. Starting with the unique researcher ID. then researcher username and password to let the researcher log in to the website and log in to their application. Also it will contain the researcher name and address if there is any mail should be sent to the researcher. Also it will contain their emails, phone number, class and which industry that they are in.



**Partnership Table**

partnership ID - int(10)

partnership name - varchar(20)

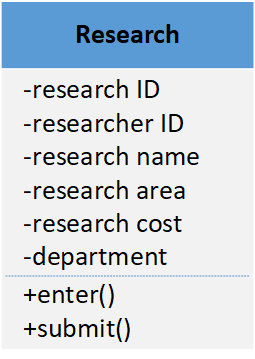
industry email - varchar(20)

research ID - int(10)

partnership description - varchar(200)

**Narrative**

Partnership table consists of partnership ID- int(10) then partnership name- varchar(20)industry email-varchar(20).research ID-int(7) as we choose in the partnership table. Partnership description-varchar(200) to give an option for any additional notes.



**Research Table**

research ID - int(10)

researcher ID - int(7)

research name - varchar(15)

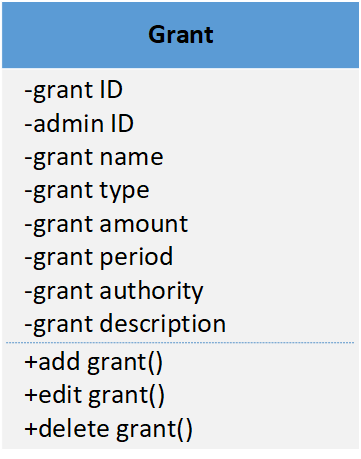
research area - varchar(20)

research cost - int(7)

department - varchar(20)

**Narrative**

The research table includes a research ID which will be unique for each research. Then researcher ID to identify the student who is participating in the research. Then information about the research starting with research name and area. Then the cost of this research and lastly the department that this research is part of.



**Grant Table**

grant ID - int(10)

admin ID - int(7)

grant name - varchar(20)

grant type - varchar(10)

grant amount - int(7)

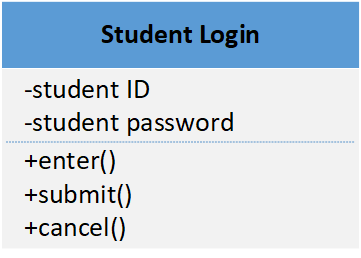
grant period - varchar(10)

grant authority - varchar(10)

grant description - varchar(200)

**Narrative**

In this table it is explaining the grant information and it also includes admin ID. The admin class holds admin’s attributes and the grant class holds grants attributes. The client after seeing this class model will get an impression of the detailed attributes and the associated method regarding the function to modify grants use case. The admin class interacts with the grants class to modify or delete the already existing grant having one to many multiplicity relations. Both classes have an association relationship with each other. In this table we have grant Id which will be unique to each grant. Admin ID as we described above. Grant name, type to identify this grant which type it will be then grant amount int(6). Grant period to specify this grant will be given in which semester. Then grant authority and description to add more details.



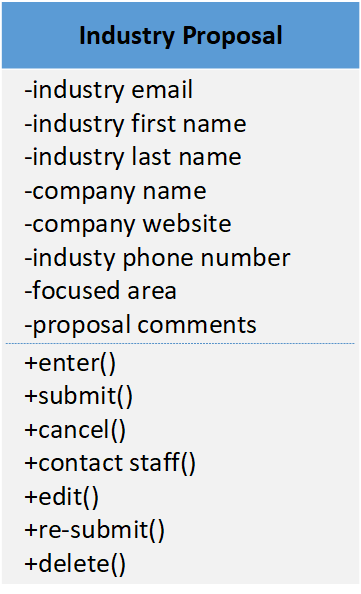
**Student Login Table**

student ID - int(7)

student password - int(20)

**Narrative**

The login table contains User ID-int(7) as a unique Id for each student and Password- int(20) attributes. The User class contains two methods, enter and submit. It interacts with the User Data class to login, and the User Data class interacts with the Roles class to distribute permissions.



**Industry Proposal Table**

industry email - int(20)

industry first name - varchar(12)

industry last name - varchar(12)

company name - varchar(20)

company website - varchar(20)

industry phone number - int(10)

focused area - int(6)

proposal comments - varchar(200)

**Narrative**

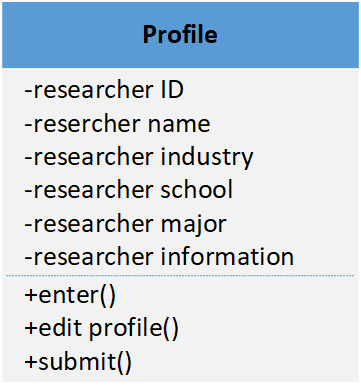
When the industry partner goes to UofL Research & Innovation website, they click on

the tab that leads them to the page designed for industry users. When the industry

partners go to the page, they will see a section dedicated for “Industry Request”, and

when they click on the button “Industry Request Form”, they will be directed to a new

page with three buttons. They can click on the create, edit or delete Industry Request Form” button. After clicking the button if they want to create or edit, they will see a box prompting them to enter the email address int(20)that is associated with the proposal. After entering the matching email address, the application is consist of industry first name-varchar(12)industry last name-varchar(12)company name-varchar(20) company website-varchar(20)industry phone number-int(10)focused area-int(6) proposal comments-varchar(200) the system traces back the proposal, then the industry representative can make changes to the industry proposal/request they have already submitted using the information they have. They click on the “Re-submit” button and the changes will be sent to the UofL Research &Innovation Office for review and stored in the UofL database.



**Profile Table**

researcher ID - int(7)

researcher name - varchar(20)

researcher industry - varchar(20)

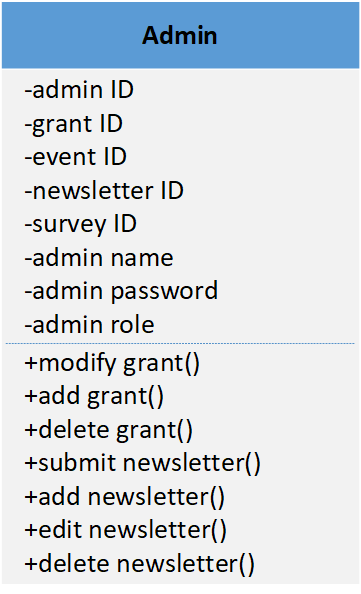
researcher school - varchar(20)

researcher major - varchar(20)

researcher information - varchar(50)

**Narrative**

In order to create a profile, then the researcher will first have to log in to the Research and Innovation website. After this they will be prompted to create their profile including basic information, and research bio researcher ID-int(7)researcher name-varchar(20)researcher industry-varchar(20) researcher school-varchar(20) researcher major-varchar(20) researcher information-varchar(50)This will then become public on the website for other researchers, industries, and administration to see.



**Admin Table**

admin ID - int(7)

grant ID - int(10)

event ID - int(10)

newsletter ID - int(10)

survey ID - int(10)

admin name - varchar(20)

admin password - varchar(20)

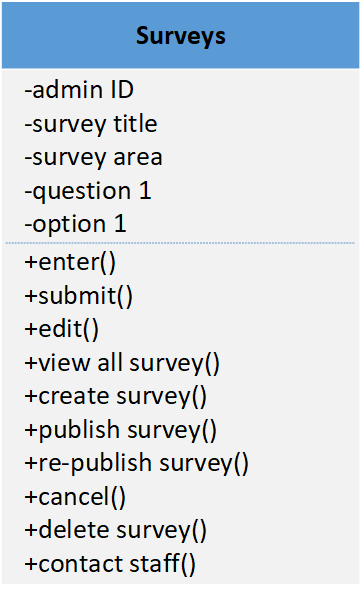
admin role - varchar(20)

**Narrative**

Admins goes to the Research & Innovation website, and click on the “Log In” tab, then

he/she will be directed to the login page. After he/she enters the designated username

and password, the database will verify the credentials and allow the admin to log in. the admin table will contain every ID that the admin has. Which in our project admin ID- int(7), grant ID-int(10), event ID-int(10), newsletter ID-int(10, )survey ID-int(10), admin name-varchar(20), admin password-varchar(20), admin role-varchar(20)



**Surveys Table**

admin ID - int(10)

survey title - varchar(20)

survey area - varchar(15)

question 1 - varchar(20)

option 1 - varchar(10)

**Narrative**

Admin goes to the Research & Innovation website, and logs in. Admin clicks on “Resource” tab, and clicks on “View All Surveys” in the Survey section. After being directed to the survey page, admin clicks on the “Create Survey” which contains survey title-varchar(20), survey area-varchar(15), question 1-varchar(20), option 1-varchar(10). Then the survey is completed, the admin clicks on the button “Publish Survey”. Admin can also modify and delete the survey.



**Event Table**

event ID - int(10)

event name - varchar(20)

event location - varchar(20)

event date - varchar(8)

event time - varchar(20)

event details - varchar(200)

**Narrative**

The Event table is a mix between the administrator class and the event class. Admin table which holds the information the system holds for each admin after the system acknowledges this information, they can therefore connect to the events class and edit the information for an event from that class. This is a one to many relationship because an admin can edit more than one event if they like. The event class contain event ID-int(10)which is unique to identify each event, event name-varchar(20), event location-varchar(20), event date-varchar(8), event time-varchar(20), event details-varchar(200).



**Newsletter Table**

newsletter ID - int(10)

newsletter title - varchar(20)

newsletter author - varchar(20)

newsletter description - varchar(200)

**Narrative**

The Newsletter table contains a newsletter ID which is unique for each newsletter. It is important to have a newsletter to keep everyone updated about the latest news. It also contains a newsletter title, newsletter author, newsletter description.



**Backup Data Table**

Data entry ID - varchar

Calendar Date - date

Data entry creation time - time

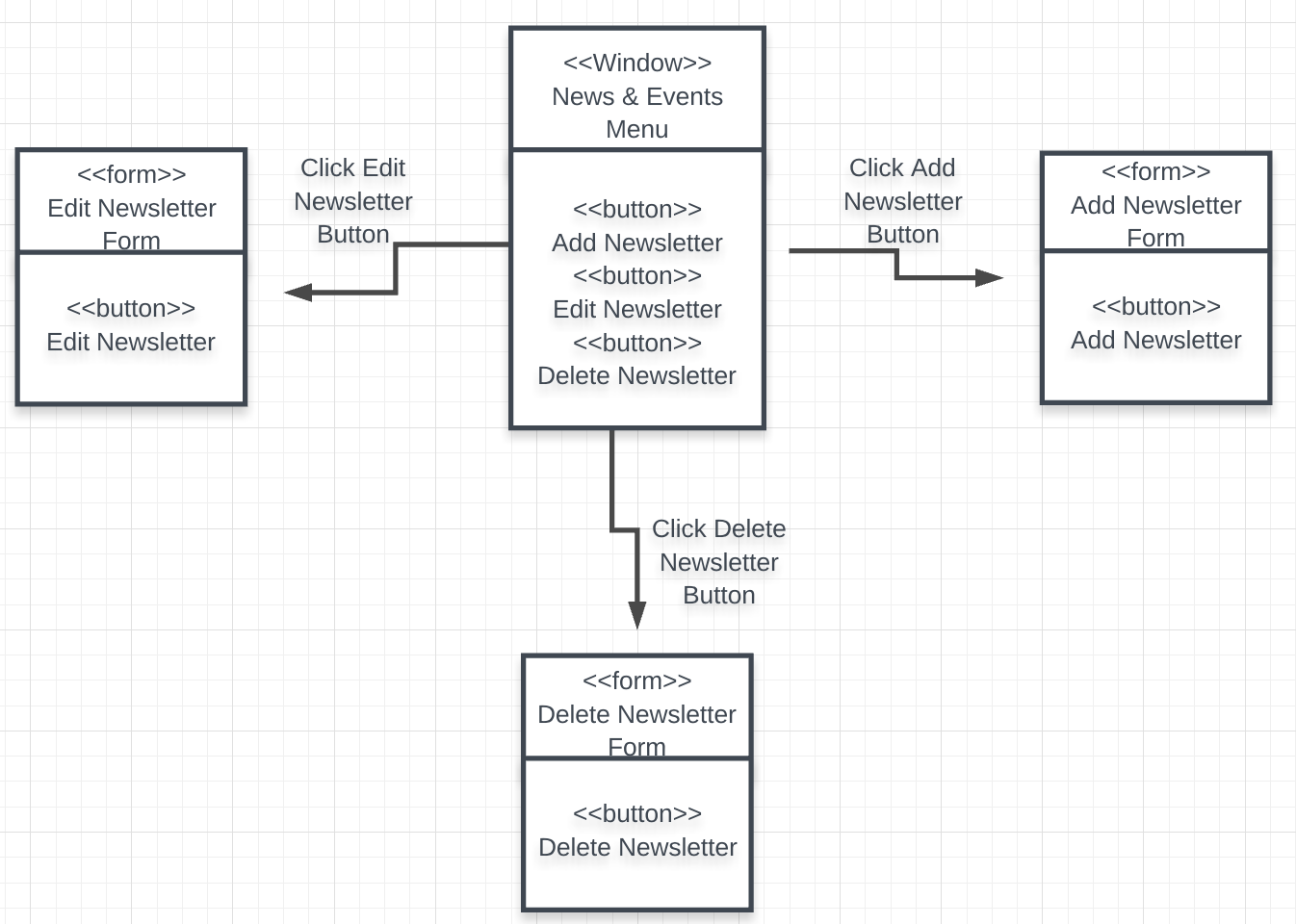
**Narrative**

This data table describes the flow of action through which the actor (admin) logs into the system and starts a data backup and restoration. The system verifies their permissions and allows them access to this functionality.

The backup data class diagram contains Data, Calendar Date, and Time attributes. The class contains one method, store data. It interacts with the restore data class because the data must be accessed for a rebuild.

The class contains one more method, restore data. It interacts with the backup data class because the data must be accessed for a rebuild.

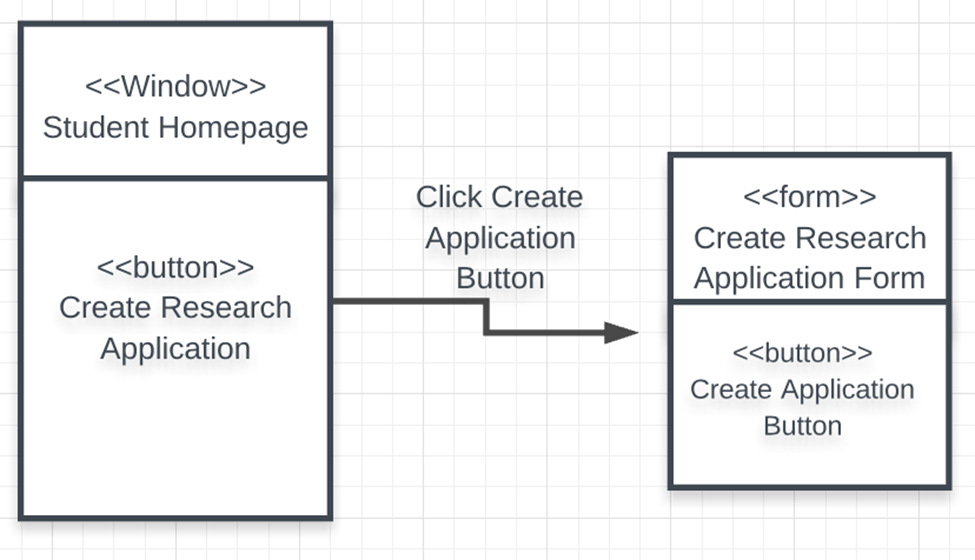
1. **User Interface Navigation Diagram and Screen Layouts**
   * The Window Navigation Diagram depicts the screens needed to realize the use cases. The screen layouts are captured from the HTML prototype and realize the use cases.



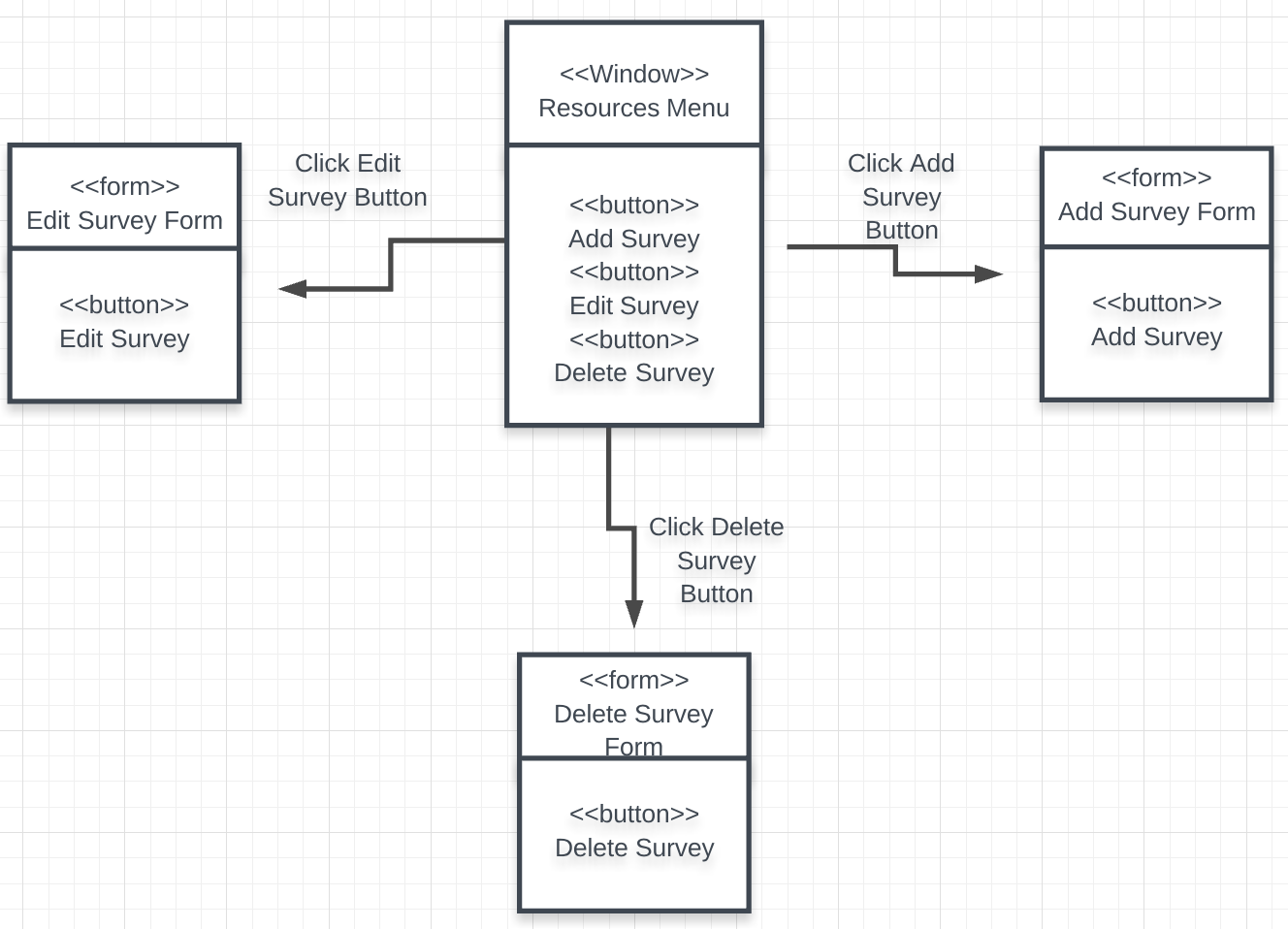
Create, Edit, and Delete a Newsletter



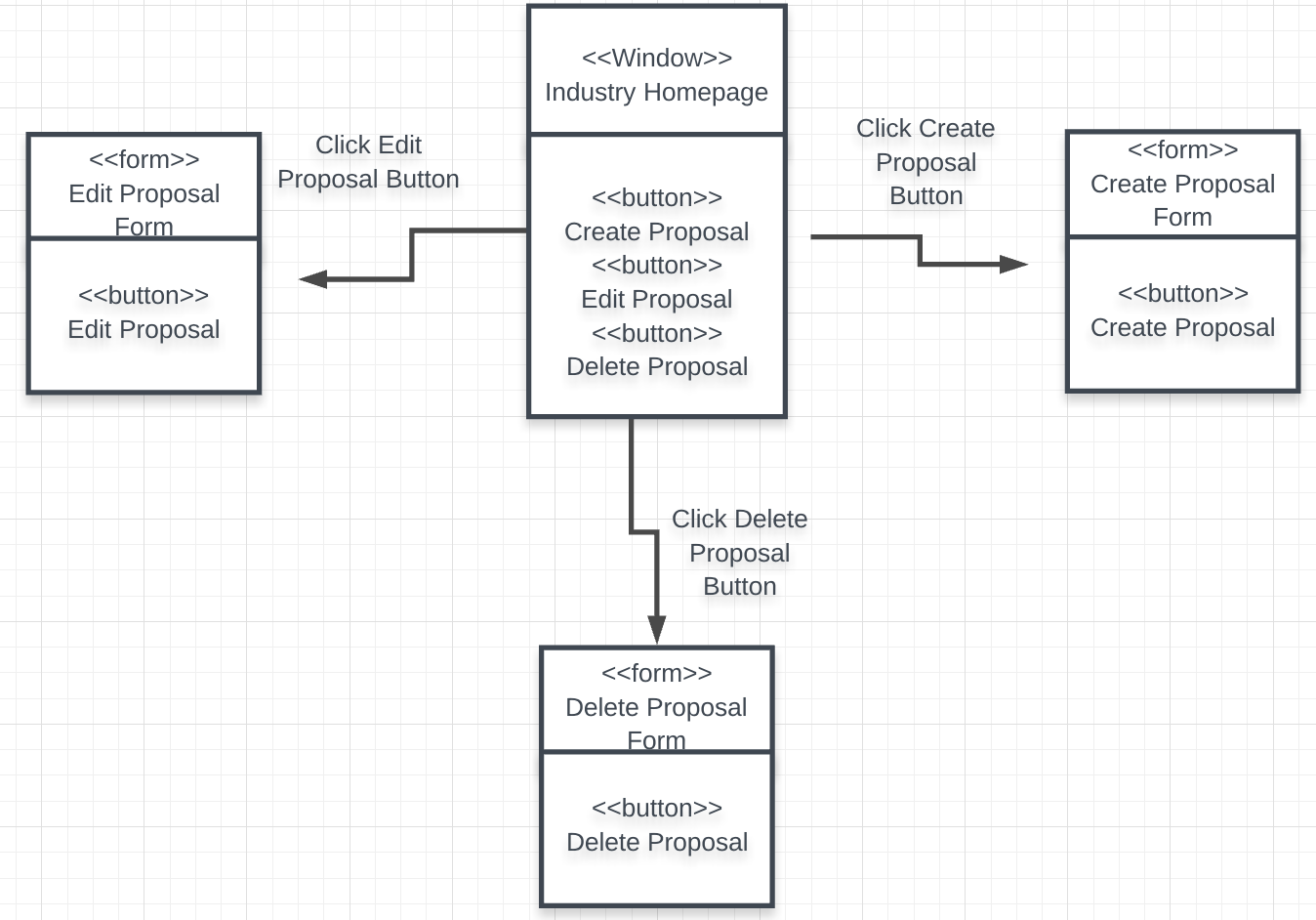
Add, Edit, and Delete an Event



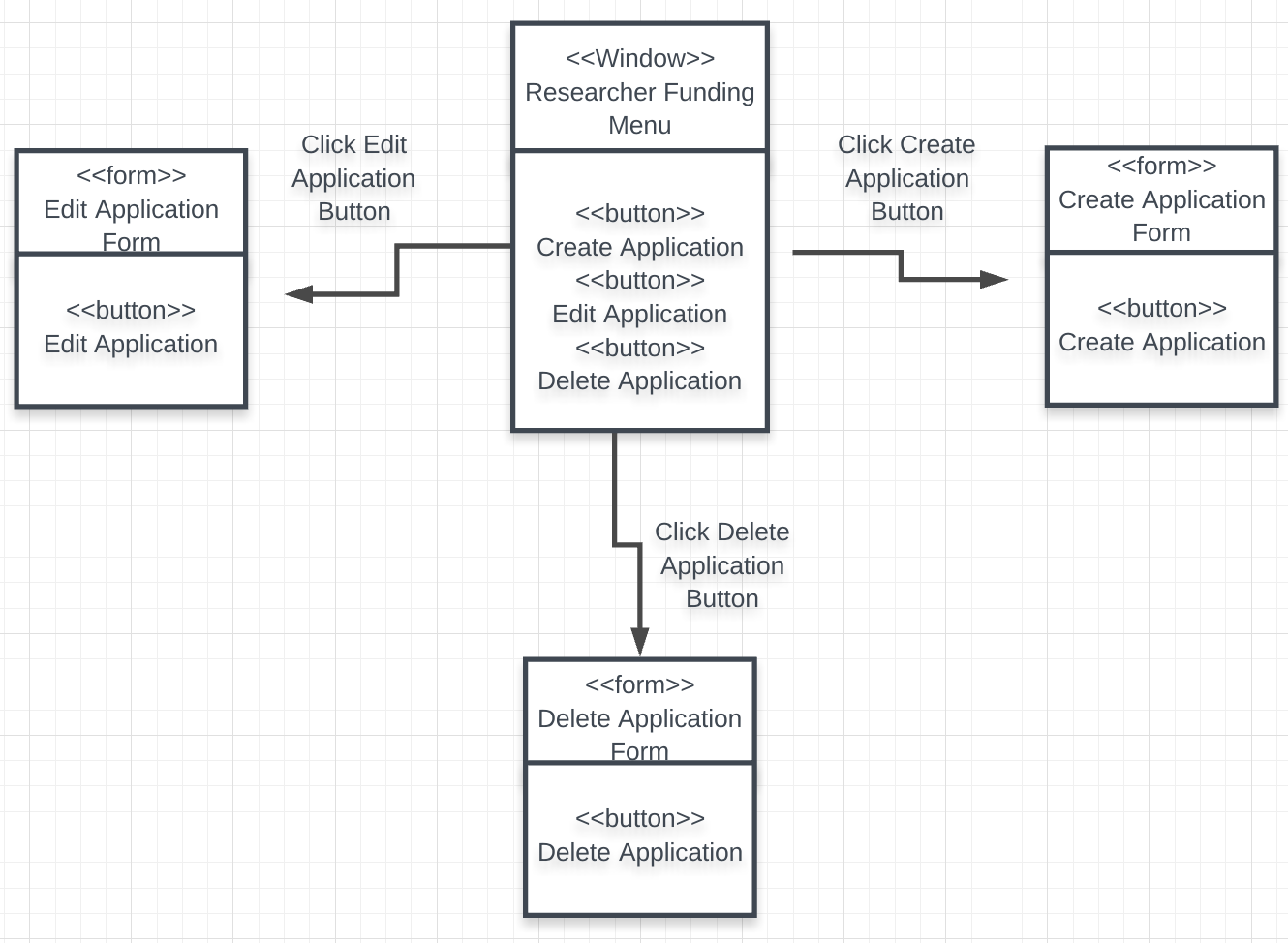
Create Research Application



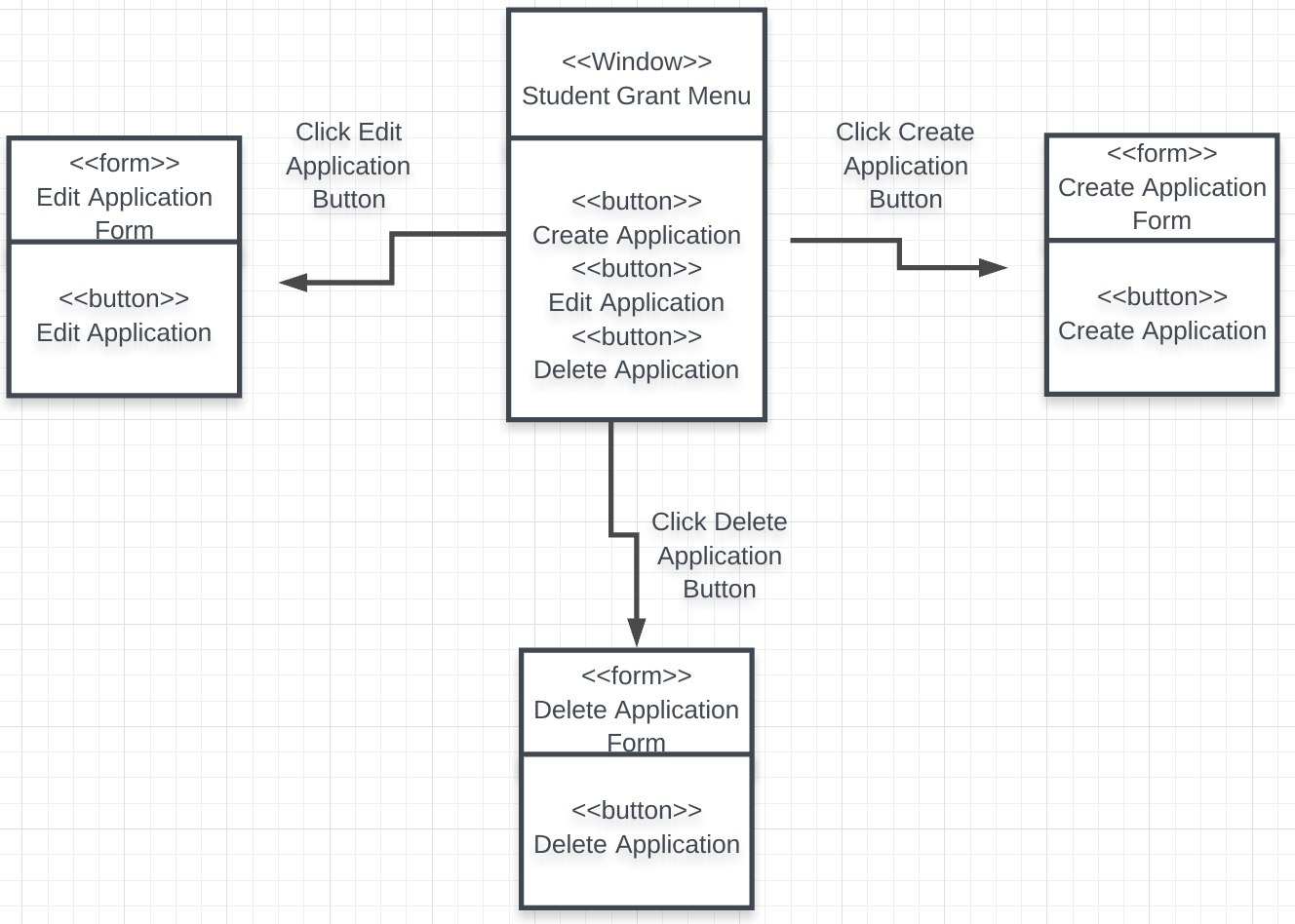
Create, Edit, and Delete a Survey

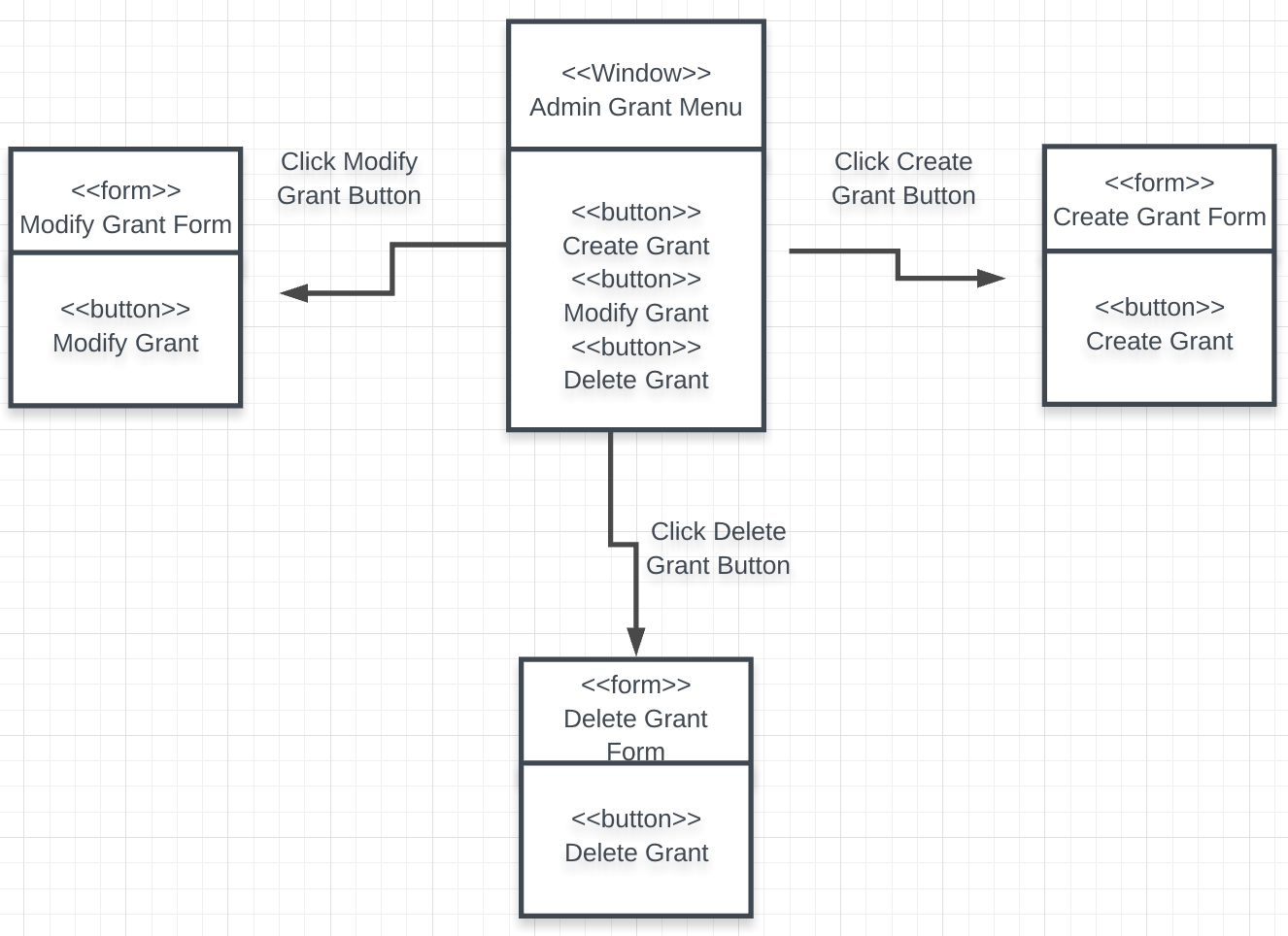


Create, Edit, and Delete a Industry Proposal

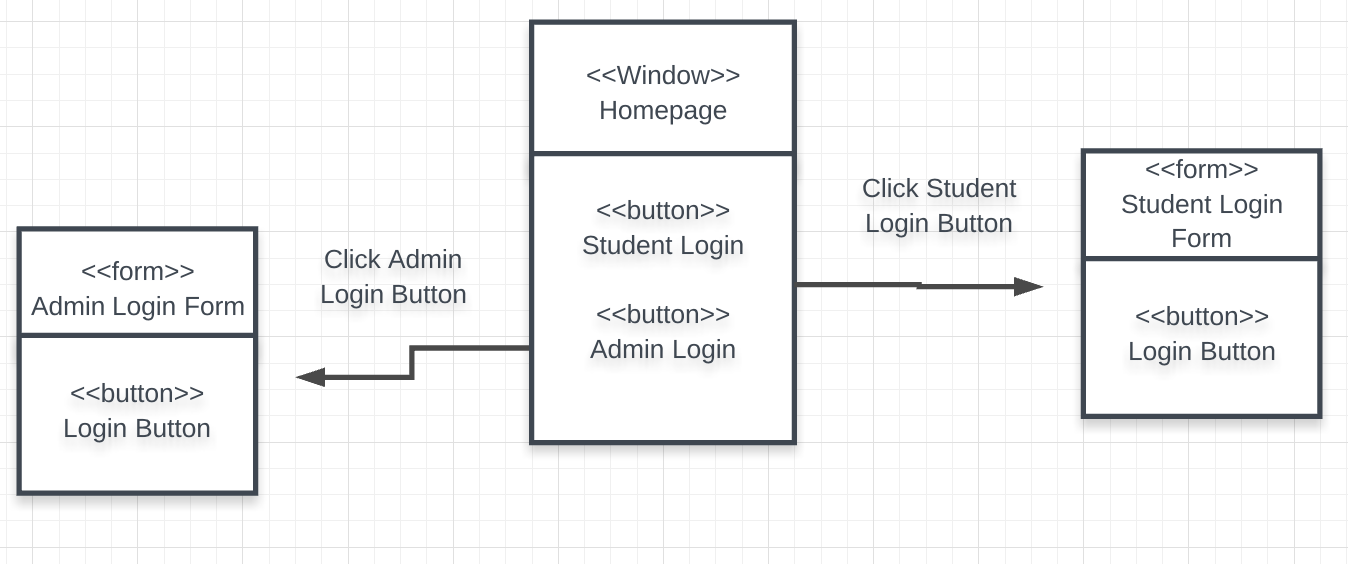


Create, Edit, and Delete a Funding Application

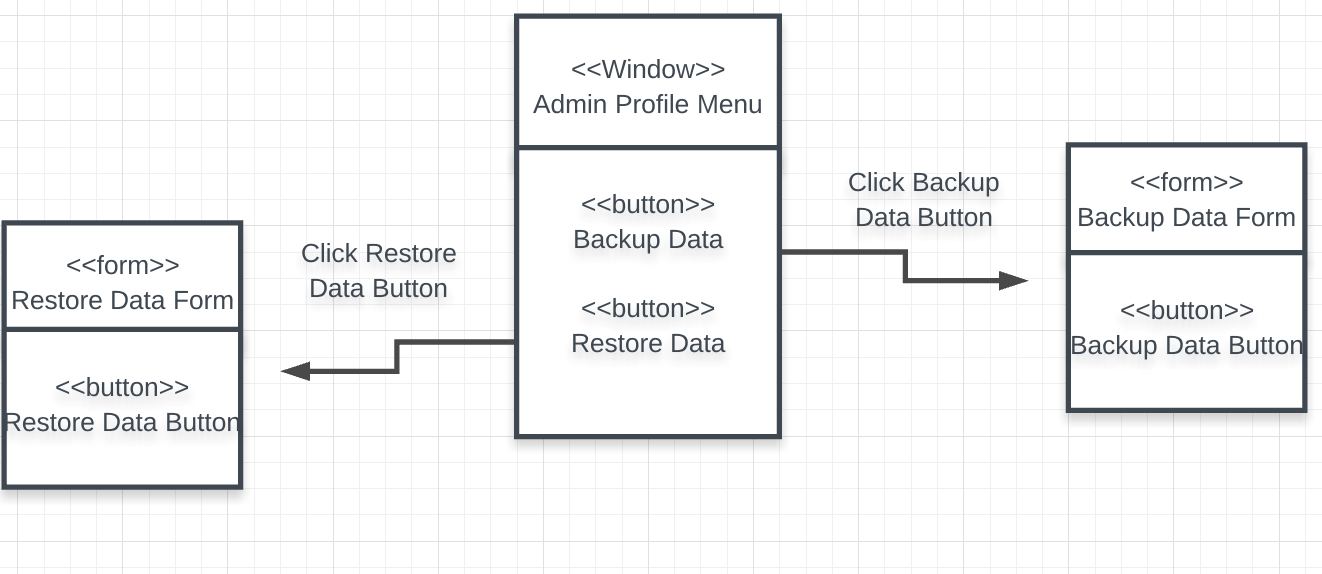


Create, Edit, and Delete a Grant Application

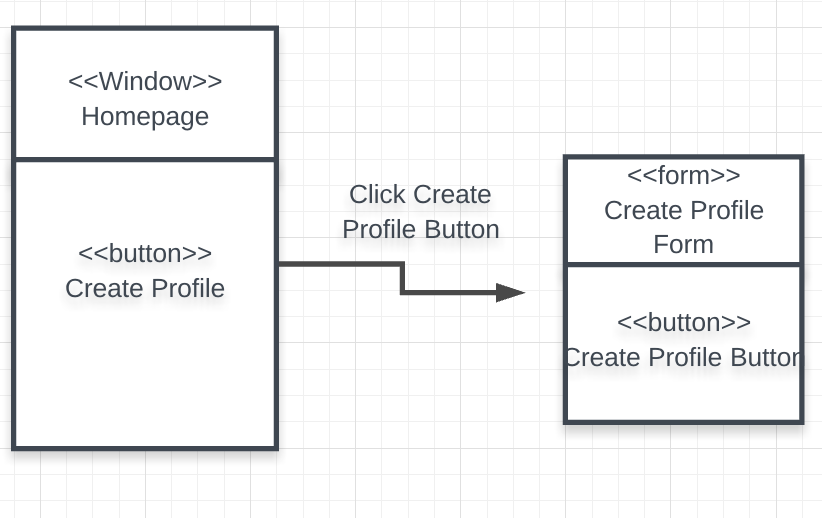
Create, Edit, and Delete a Grant



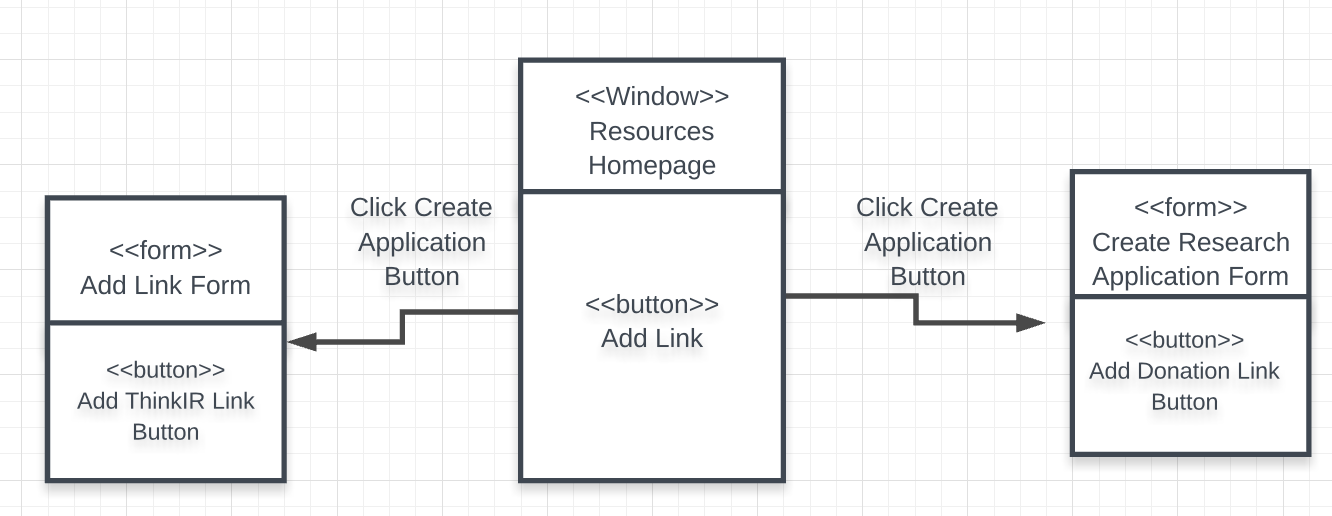
User Login



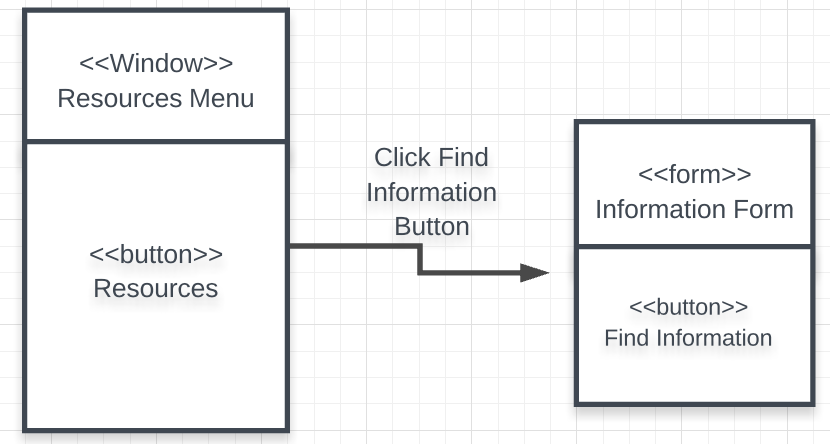
Backup and Restore Data



Create Profile



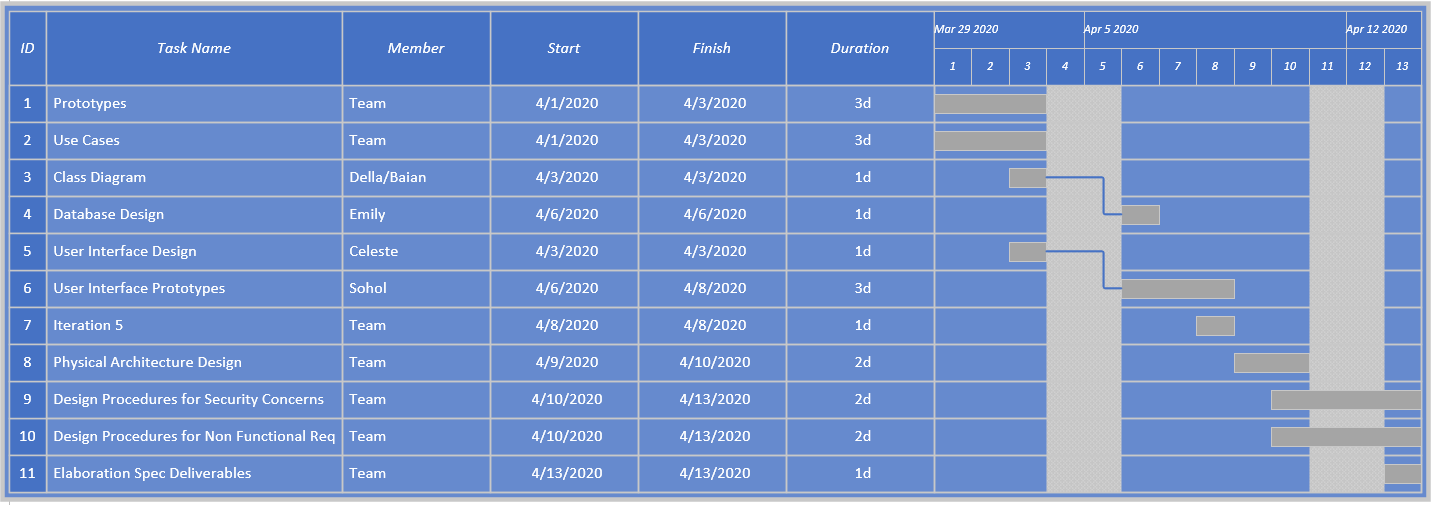
Donation and ThinkIR Links



Find Grant Information

|  |
| --- |
| **UC 1 – Create Research Application**   * Students can find the research application module which includes a link to the form. |
| **UC 2 – Update Information on Current Partnerships**   * Admins can find this module under the For Industry tab. After locating the Current Partnerships module, they will be able to update information via a form. |
| UC 3 – Add Weekly Updates to Newsletter   * Admins can find this module under the News and Events tab. After locating the Newsletter module, they will be able to create a newsletter via a form. |
| **UC 4 – Delete Newsletter**   * Admins can find this module under the News and Events tab. After locating the Newsletter module, they will be able to delete a newsletter. |
| **UC 5 – Edit Newsletter**   * Admins can find this module under the News and Events tab. After locating the Newsletter module, they will be able to edit a newsletter. |
| **UC 6 – Find Information**   * The student can find this module under the For Students tab. Clicking the reroute link opens a form that allows the student to search for grants. |
| **UC 7 – Create Grants**   * The admin can find this module under the For Researchers tab. Clicking the link reroutes them to a form that allows them to create a grant. |
| **UC 8 – Modify Grants**   * The admin can find this module when logging in. Clicking the link reroutes them to a form that allows them to modify a grant. |
| **UC 9 – Delete Grants**   * The admin can find this module under the For Researchers tab. Clicking the link reroutes them to a form that allows them to delete a grant. |
| **UC 10 – Create Grant Application**   * The student can find this module under the For Students tab. Clicking the link reroutes them to a form that allows them to create a student application. |
| **UC 11 – Edit Grant Application**   * The student can find this module under the For Students tab. Clicking the link reroutes them to a form that allows them to edit a student application. |
| **UC 12 – Delete Grant Application**   * The student can find this module under the For Students tab. Clicking the link reroutes them to a form that allows them to delete a student application. |
| **UC 13 – Create Funding Application**   * The researcher can find this module under the For Researchers tab. Clicking the link reroutes them to a form that allows them to create a funding application. |
| **UC 14 – Edit Funding Application**   * The researcher can find this module under the For Researchers tab. Clicking the link reroutes them to a form that allows them to edit a funding application. |
| **UC 15 – Delete Funding Application**   * The researcher can find this module under the For Researchers tab. Clicking the link reroutes them to a form that allows them to delete a funding application. |
| **UC 16 – Create Industry Proposal**   * The industry partner can find this module under the For Industry tab. Clicking the link reroutes them to a form that allows them to create an industry proposal. |
| **UC 17 – Edit Industry Proposal**   * The industry partner can find this module under the For Industry tab. Clicking the link reroutes them to a form that allows them to edit an industry proposal. |
| **UC 18 – Delete Industry Proposal**   * The industry partner can find this module under the For Industry tab. Clicking the link reroutes them to a form that allows them to delete an industry proposal. |
| **UC 19 – Add Surveys**   * The admin can find this module under the Resources tab. Clicking the link reroutes them to a form that allows them to create a survey. |
| **UC 20 – Edit Survey**   * The admin can find this module under the Resources tab. Clicking the link reroutes them to a page that gives them the option to edit or delete a survey, when edit is clicked, a form pops up that allows them to edit it. |
| **UC 21 – Delete Survey**   * The admin can find this module under the Resources tab. Clicking the link reroutes them to a page that gives them the option to edit or delete a survey. |
| **UC 22 – Log In**   * When clicking the log in tab, the user can access the Log In module. A link titled Log In Here takes them to a form. When clicking other links throughout the website, an user that is not logged in will be prompted to log in before accessing information. |
| **UC 23 – Backup Data**   * An admin can access the Admin Resources module under the Log In tab. Clicking the link takes them to a form that will start the backup process. |
| **UC 24 – Restore Data**   * An admin can access the Admin Resources module under the Log In tab. Clicking the link takes them to a form that will prompt them to start the restoration process by choosing a backup. |
| **UC 25 – Create Profiles**   * A researcher will be prompted to create a profile when they log in for the first time, anytime after that they will be able to update it. They can find this functionality in the Log In module under the Log In tab. They will be able to create a profile to showcase their work. |
| **UC 26 – Add Event**   * The admin can find the Events module under the News and Events tab. Clicking the link reroutes them to a page that gives them the option to create, edit, or delete an event, when New Event is clicked, a form will pop up. |
| **UC 27 – Edit Event**   * The admin can find the Events module under the News and Events tab. Clicking the link reroutes them to a page that gives them the option to create, edit, or delete an event, when Edit Event is clicked, a form will pop up. |
| **UC 28 – Delete Event**   * The admin can find the Events module under the News and Events tab. Clicking the link reroutes them to a page that gives them the option to create, edit, or delete an event, when Delete Event is clicked, a form will pop up. |
| **UC 29 – Link The Donation Process Directly to UofL Development Departments**   * User will be able to donate directly to the university when clicking the donation link found under the resources tab. It will reroute them to the University’s development department’s website. |
| **UC 30 – Link ThinkIR to the Research Page**   * Users will be able to access ThinkIR after clicking its link under the For Researchers tab. It will reroute them to the ThinkIR website so they can log in. |

1. **Gantt Chart**
   * A Gantt chart is a type of bar chart that illustrates a project schedule. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis. The width of the horizontal bars in the graph shows the duration of each activity.



1. **Prototype**
   * A mock-up/demo of what the website will look like when it goes live. A fully functional HTML website that realizes all of the use cases presented in previous iterations.
   * Link: <http://pictureintext.net/pro/>

|  |
| --- |
| Home Page   * The home page provides four branches for different types of users, it directs them to pages with unique and relevant information. * The search function helps users search for information they need across the entire website. * On the bottom, it links the users to Research Office’s social media accounts, physical locations, and contact information. |
| For Student Page   * The page is dedicated to the students and provides them with information about undergraduate research scholar grants, internal and external research opportunities, and other resources. |