COMS3002 Software Engineering

Project 11 - Postgraduate Application Approval System

WITS
UNIVERSITY

Group 8
Abdulkadir Dere - 752817
Jesse Wright - 721386
Liam Leibrandt - 814078
Brenda Lin - 747243

School of Computer Science
University of Witwatersrand
2 October 2017

Contents

1 Introduction

1.1 Glossary

Term/Acronym/Abbreviation	Description/Definition
PGO	Postgraduate Officer
PGC	Postgraduate Coordinator
PGFO	Postgraduate Faculty Officer
Evaluator	Person who is responsible of evaluating the application for the recommendation phase.
EIE	The School of Electrical and Information Engineering
PAAS	Postgraduate Application Approval System
SIMS	Students Information Management System
Applicant	User who registers on the system to apply (formal request) for postgraduate degree
Application	Formal request submitted by the applicant to apply for a postgraduate degree
Associated Documentation	Any documentation that is associated with the application form. Retrieved from SIMS. These documents may also need to be analysed with the application by the users.
CRUD	Create, Read (View), Update (Edit) and Delete (Archive). Used to manage entities in the system
UX	User Experience
MVC	Model View Controller
IIS	Internet Information Service

1.2 Problem Statement and Project Purpose

The purpose of the Postgraduate Application Approval System is to provide an efficient, paperless, online application system to the school of Electical and Information Engineering at the University of Witwatersrand. The current system is a paper based system, dependent on manually passing application documents between staff and student alike. This process can be inefficient, unreliable as well as taxing on the environment in paper usage, making the application process highly unappealing and cumbersome to prospective students. The PAAS system is a much needed solution and will rectify any such dependencies that may have a negative affect on the application process.

1.3 Project Overview

Our aim for the project is to create an online postgraduate application approval system for the school of electrical and information engineering. This system will be completely paperless to keep the paperwork of the activity to a minimum. The PGO will receive completed applications from students . These applications will be checked to make sure they are ready to process. Once they are checked, the applications with the required information can be sent to one of the three users that will either recommend or not recommend an application. The three actors are the Research Group Lead, Identified Supervisor and the PGC. The application will be sent to either one of these actors based on the program that the application is for. If an interview is needed, one of the three actors can book an interview with the applicant. After the interview the user can recommend/not recommend the application. The application will then be sent to the PGC who will then accept or reject the application based on the the application being recommended or not and based on faculty rules and regulations. The PGFO will receive an email/notification about the application's status. The applicant and the schools PGO will also receive an email notifying them whether the application was accepted or rejected with an explanation.

1.4 Summary of Benefits

- Automates the application approval system.
- Reduces delays for the application process.
- Environmentally friendly.
- Keeps track of the process.
- Stores the application and other relevant data digitally.
- Eliminates loss of documents.

2 Software Requirement Specification

2.1 Overall Description

2.1.1 Product Perspective

The solution we are developing will be a web application. This web application will be used by the employees of the EIE who are responsible for the postgraduate approval process of applicants to their graduate program. Our solution, the Postgraduate Application Approval System (PAAS), will provide these employees with an almost completely paperless electronic way of approving postgraduate applicants.
The PAAS will be designed to:

- Send notification emails to PGO about applications that need to be processed.
- Receive and view applications and associated documents.
- Forward documents to Evaluator (Research Group Leads, Identified Supervisors or PGCs).
- Create interviews for applicant and notify them by email.
- Allow applications to be recommended by Evaluators.
- Send application to PGC.
- Allow PGC to accept/decline application.
- Send the accepted/declined applications back to PGO.
- Send notification email to PGFO.
- Send email to applicant whether he/she has been accepted.
- Print documents if needed at any time.
- Login users.
- Create users.
- Update users if needed.

2.1.2 Requirements Gathering

Brainstorming: We got together as a group and identifying as many possible solutions to the problem that the EIE is facing. We then simplified the solution details. Brainstorming helps casts a broad net, determining various discrete possibilities. Then simplifying and prioritizing the details of the solution. [2]

Observation: We were given a step-by-step walkthrough of the business process, which we believe is a more subjective form of obtaining requirements than pure observation. We then took those steps and converted them into functions for the PAAS. [2]

2.1.3 Use Cases

We will be converting what the PAAS is designed to do into use cases.
Main Use Case List:

- Create Application
- Read Document
- Create Interview
- Recommend Application
- Accept Application
- Login User

Secondary Use Case List:

- Print Document

CRUD (Create, Read [View], Update [Edit], Delete [Archive]) Use Case List:

- ie. Manage PGO = Create PGO, View PGO, Update PGO, Archive PGO
- Manage PGO
- Manage PGC
- Manage PGFO
- Manage Evaluator (Research Group Lead or Identified Supervisor)
- Manage Application
- Manage Interview
- Manage Document

2.1.4 User Characteristics

The users are the people and other systems that interact with the PAAS system. A user can be primary user or a secondary user. A primary user interacts directly with the PAAS and a secondary user interacts with the PAAS indirectly.

User List:

User	Primary/Secondary	Interaction with PAAS
PGO	Primary	• Receives email from PAAS about applications for processing. • Gets redirected to SIMS. • View applications and associated documents. • Forward documents to Evaluators. • Send notification email to PGFO. • Ability to print documents.
Evaluator	Primary	• Receive documents from PGO. • View applications and associated documents. • Setup applicant interviews. • Recommend/Don't recommend application. • Send documents to PGC and PGO. • Ability to print documents.
PGC	Primary	• Receive documents from PGO and Evaluators. • View applications and associated documents. • Accept/Reject application. • Send documents to PGO. • Ability to print documents.
PGFO	Secondary	• Receive email notifications from PGO. • Ability to print documents.
Applicant	Primary	• Receives interview emails. • Receives email about application status.
Emailing System	Secondary	• Sends notification emails to the relevant actors.

2.1.5 General Constraints

Implementation
Not all internet browsers may work with our system. Moving from manual to digital may be time consuming, and are subject to human error. The number of active users may start out small due to human resistance towards new technology, especially those who are not computer savvy. Teaching new users how to use the system will be time-consuming.

Due to time constraints and the fact that we are students, the system may not be fully-functional as planned.

Hardware
Any device that makes use of a supported browser will be able to use the system. We cannot guarantee that all devices will be supported.

The system will require an internet connection.

Software
One needs a supported browser. The application will be available to access from computer and mobile devices using a browser.

Legal Issues
To obtain a web domain. The source code will belong to the University and therefore, if the client wants the rights to the source code, they might have to go through legal protocols to obtain it from Wits University.

As students we may not be given permission to access SIMS.

Reliability and Fault Tolerance
The system needs to be reliable and should be able to recover the student documents. It is extremely frustrating for applicants to re-upload applications because of the unreliability of the system.

The system also needs to have as little faults as possible, since we are working with an important process at the university, this process cannot be put on hold because of a faulty system.

Security
The system is working with sensitive information and cannot be compromised. Student details and marks are very private pieces of data and cannot be leaked because of a poorly designed system.

User
Based on the security issue mentioned above, users will only be able to access the system with a username and password. Therefore users should not have access to other users' data.

The PGO should not have access to make the final decision until the recommendation for the application is received from the relevant users.

2.1.6 Assumptions and Dependencies

- We are assuming all users have a supported browser.
- We are assuming that all applicants are Wits students (because of time constraints we are not regarding non-Wits student applicants).
- We are assuming all users are computer literate.
- The system will be dependent on a local database.
- We are assuming that all applicants and users use email actively.
- We are assuming that all users may need to print the application documents.
- We are assuming that PGO is in charge of creating users.

2.2 Detailed Requirements

2.2.1 External Interface Requirements

Interfaces
The user interfaces may be different depending on what type of user is logged into the system. But all interfaces will follow some fundamental UX principles. Some of these UX principles are digestibility, clarity, trust, familiarity and delight. Digestibility gives the user the feeling of I get it. The format, components and layout of the interface should be as clear as possible so that the user can have a feeling knowing exactly what to do because of past experiences and familiarity. Clarity is used in terms of the components, fields, layout, validation, error messages and format. The formats, validation and error messages have to be clear in terms of language, ie. the field requires a valid email address. A user should never feel unsure when entering their details. The use of components such as date-time picker gives the user a feeling clarity and trust. The users of the PAAS should have a feeling of familiarity from the previous forms that used to fill in manually. The electronic forms should be designed around the manual forms, the formats and positions need to be as similar as possible to allow for an easier transition. A user should have a feeling of delight when using the system, they should never feel frustrated because this will lead to the users being reluctant to using the system. [1]

Hardware Interfaces
Since this solution is a web-based application, the hardware devices used must support the use of web browsers, as well as the ability to display a GUI and process input from the user in order to perform the interactions between client and server. To display the GUI of the application, a display device must be used, preferably with a DPI (dots per inch) above 300. If the DPI of the device is too low, the GUI may be too pixelated to view or give meaning to. For input, a keyboard is required. It may be a digitally displayed keyboard (on the display of a device) or a physical external keyboard. The keyboard is required for basic functionality of the application. Also on the aspect of input, a mouse or trackpad is required in order to perform basic mouse down functions as well as cursor movement. The device must have sufficient processing power and memory in order to run the web browser which will be the host of the web application on the device.

Software Interfaces
The software used for this web-based application will be web browsers. The web browsers which this applications functionality will be tested on are FireFox, Google Chrome, Microsoft Edge and the mobile versions of these. As discussed above, the hardware devices need to be able to support FireFox, Microsoft Edge and Google Chrome web browsers.

Communication Interfaces
The system will make use of email functionality to notify the users, both primary and secondary. The email function is used to notify applicants about the status about their application. The PGO will receive emails when there are new applications to be processed.

2.2.2 Functional Requirements

Use Case 1:	Create Application
Primary Actor:	Applicant
Precondition:	1. The applicant must exist in the database. 2. The applicant must be logged in.
Main Success Scenario:	1. The user will request to create a new application for the system. 2. The system will prompt the user to enter student number, first and last name, ID number, email, contact number, school, faculty, street number, street name, suburb, city, province. 3. The system will prompt the user to confirm the selection. 4. The user will confirm. 5. The system will prompt the user to upload their documentation. 6. The system will prompt the user to confirm the selection. 7. The user will confirm. 8. The system will notify the user with a success message that the user has successfully created an application.
Exception Scenarios:	1. The student number exists, if so the user will be displayed a message indicating that they have already registered. 2. An error message will be displayed and system will redirect the user to the home page.

Use Case 2:	Read Document
Primary Actor:	PGO, Evaluator, PGC
Precondition:	1. The user must be logged in. 2. The user must have an application that needs to be processed and/or the associated documents.
Main Success Scenario:	1. The user will prompt the system to view a certain document. 2. The system will open the document to be viewed.
Exception Scenarios:	None.

Use Case 3:	Create Interview
Primary Actor:	Evaluator (any relevant actors - PGC, research group lead or identified supervisor)
Secondary Actor:	Applicant
Precondition:	1. The user must be logged in. 2. The user must have an application that needs to be processed. 3. The application should not be evaluated.
Main Success Scenario:	1. The user will prompt the system that they want to setup an interview with the applicant. 2. The system will open the interview form. 3. The system will prompt the user to enter in the details of the interview such as a date, time and venue. 4. The user will fill in these details. 5. The system will prompt the user to confirm the interview details. 6. The user will confirm the details. 7. The system will notify the user that the interview creation was successful. 8. The system will redirect the user to the home page.
Exception Scenarios:	1. The user will not confirm the details of the interview. 2. The system will keep the details intact, since the user could have made a small mistake that the user needs to change. 1. The system will notify the user that the interview creation was not successful. 2. The system will redirect the user back to the interview form.

Use Case 4:	Recommend Application
Primary Actor:	Evaluator, PGC
Precondition:	1. The user must be logged in. 2. The user must have an application that needs to be processed. 3. The application should not be evaluated.
Main Success Scenario:	1. The user will prompt the system to a recommend an application. 2. The user will enter the recommendation description. 3. The system will prompt the user to confirm the recommendation. 4. The user will confirm the recommendation. 5. The system will send the application to PGC for final decision.
Exception Scenarios:	1. The user will not confirm the recommendation. 2. The system will redirect back to the previous screen.

Use Case 5:	Finalize Application
Primary Actor:	PGC
Precondition:	1. The user must be logged in. 2. The user must have an application that needs to be processed. 3. The application should be evaluated. 4. The application should not be accepted/rejected.
Main Success Scenario:	1. The user will prompt the system to accept/reject an application. 2. The system will prompt the user to confirm the acception/rejection. 3. The user will confirm the acception/rejection.
Exception Scenarios:	None.

Use Case 6:	Print Document
Primary Actor:	PGO, Evaluator, PGC, PGFO
Precondition:	1. The user must be logged in. 2. The user must have an application that needs to be processed and/or the associated documents.
Main Success Scenario:	1. The user will prompt the system to print a certain document. 2. The system will print the document.
Exception Scenarios:	None.

Use Case 7:	Login User
Primary Actor:	PGO, Evaluator, PGC, PGFO
Precondition:	The user cannot be logged in.
Main Success Scenario:	1. The user will prompt the system to log in. 2. The system will prompt the user to enter in the email address and password. 3. The user will enter these details and login. 4. The system will direct the user to the home page.
Exception Scenarios:	1. The user will enter in the incorrect details. 2. The system will show an error message and prompt the user to enter the details correctly. 3. The user will enter these details and login. 4. The system will direct the user to the home page.

Use Case 8:	Create User
Primary Actor:	PGO
Precondition:	The user must be logged in.
Main Success Scenario:	1. The user will request to create a new user for the system. 2. The system will prompt the user to insert first name, last name, email, ID number, contact number and password for the new user. 3. The system will prompt the user to confirm the selection. 4. The user will confirm. 5. The system will notify the user with a success message that the user has been successfully created.
Exception Scenarios:	1. The the first name, last name, email, ID number, contact number or password is invalid (blank). 2. An error message will be displayed and system will redirect the user to the previous page. 1. The email has already been used. 2. An error message will be displayed and system will redirect the user to the previous page. 1. The user will not confirm the selection. 2. The system will redirect the user to the previous page.

Use Case 9:	View User
Primary Actor:	PGO
Precondition:	The user must be logged in.
Main Success Scenario:	1. The user will request to view a user in the system. 2. The system will prompt the user to select a user. 3. The system will display the user details.
Exception Scenarios:	None.

Use Case 10:	Update User
Primary Actor:	PGO, Evaluator, PGC, PGFO
Precondition:	The user must be logged in.
Main Success Scenario:	1. The user will request to update their profile. 2. The system will display the user in the system and allow the user to edit the users attributes. 3. The user will enter in the details that they want to change. 4. The user will confirm these changes. 5. The system will notify the user with a success message that the user has been successfully updated.
Exception Scenarios:	None.

Use Case 11:	Archive User
Primary Actor:	PGO
Precondition:	The user must be logged in.
Main Success Scenario:	1. The user will request to archive (delete) a user in the system. 2. The system will prompt the user to select a user. 3. The system will notify the user with a success message that the user has been successfully archived (Deleted).
Exception Scenarios:	None.

2.2.3 Performance Requirements

This application is more dependent on the accuracy of communication and information than it is dependent on the overall performance and speed of the application. We aim to have a reliable platform on which communication is priority. Although performance is not negligible, it is not a requirement at the expense of a loss of accuracy in the communication between applicant and PGO. The application will not be demanding on the hardware but it will be demanding on the bandwidth available to the device. The slower the connection to the server where the database is stored, the slower the overall interaction with the web app will be.

2.2.4 Design Constraints

The application will not be optimized for systems that run Safari browsers since there are no such devices available to our team for testing. Running the web application on a safari browser is not recommended and will be done at the user's own risk.

2.2.5 Software System Attributes

Availability: The system will be running constantly.

Security: The system's security and reliability is mentioned in Section 2.1.5

Maintainability: PGO will be admin and be able to create and archive users.
After testing and feedback, the developers would be able to update the system to suit the user's needs.

3 Design

3.1 Choice of a Software Development Life-Cycle

3.1.1 SCRUM

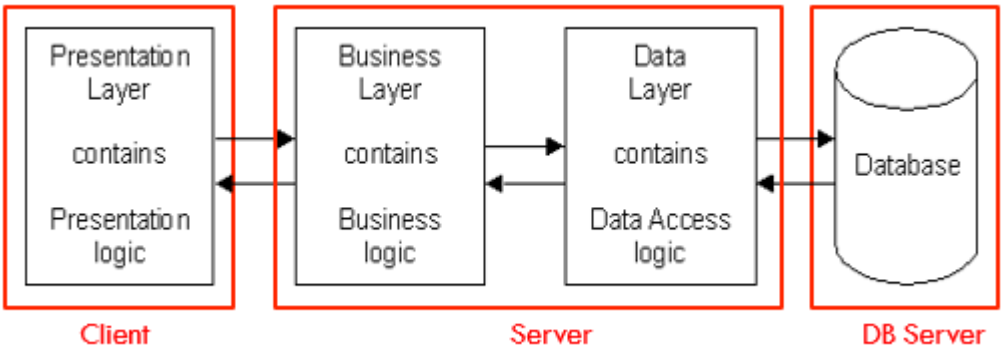
SCRUM is our choice of a software development life-cycle for our project. It is an agile development method which is iterative and incremental. This is how we plan to implement it:

- We will create a wish list of use cases and add them to our backlog.
- During sprint planning, we will pull some of the use cases from the backlog and add them to our sprint backlog, and then decide how to implement those use cases. Our sprint time is 6 - 8 weeks, depending on the team's availability.
- Along the way, the ScrumMaster (Project Leader) keeps the team focused on its goal.
- At the end of the sprint, the use cases should be implemented and work to the best of its ability.
- The sprint ends with a sprint review and retrospective. As the next sprint begins, we will choose more use cases from the backlog and begin working again. [2]

3.2 Choice of Architecture

3.2.1 Three Tier Architecture

A three-tier architecture is a programming model that enables the distribution of application functionality across three independent systems. [4]



- A Presentation Layer that sends content to browsers in the form of HTML/JS/CSS.
- An Application Layer that uses an application server and processes the business logic for the application. This might be written in C# or JavaScript.
- A Data Layer which is a database management system that provides access to application data. This will be Microsoft SQL Server (IIS Server).

3.2.2 Benefits

- It gives you the ability to update the technology stack of one tier, without impacting other areas of the application. It allows for team members to each work on their own areas of expertise.
- You are able to scale the application up and out. A separate back-end tier, for example, allows you to deploy to a variety of databases instead of being locked into one particular technology. It also allows you to scale up by adding multiple web servers.
- It adds reliability and more independence of the underlying servers or services.

- It provides an ease of maintenance of the code base, managing presentation code and business logic separately, so that a change to business logic, for example, does not impact the presentation layer. [3]

3.3 Front-end Interface Method

A web application which allows for browser support will be created. The application should then work on browsers including mobile browsers.

3.4 Back-End Service

ASP.net MVC uses SQL Server and we will use a local database (IIS Server) to store out details.

3.5 Software Requirements for the Web Application

The following software tools will be used:

- Visual Studio 2017
- SQL Server 2014
- .Net Framework 4.6.1

Please note:

- These are the applications used in the production of the Web Application. The Web Application has been tested on these software with the specified versions. There may be configuration errors if the correct versions are not used.
- MVC package must be included in Visual Studio 2017.
- We strongly advise to run the application using Google Chrome.

3.6 Other Supporting Software

Bootstrap will also be used to make sure that the Web Application has a consistent theme and is compatible on all browsers.

3.7 Student Responsibilities

- Abdulkadir Dere - Group Leader
- Brenda Lin - Quality Assurance
- Jesse Wright - Technical Lead
- Liam Leibrandt - Analysis Lead

3.8 Sprint Plan

SPRINT 1 - DATE (7 AUGUST - 20 AUGUST)

BOARD - 7 AUGUST

TO DO	IN PROGRESS	DONE	VERFICATION
Task 1 - A brief understanding of the project			
Task 2 - A meeting with the client so we can talk through the requirement specifications			

BOARD - 17 AUGUST

TO DO	IN PROGRESS	DONE	VERFICATION
		Task 1 - A brief understanding of the project (done by Abdulkadir Dere)	
		Task 2 - A meeting with the client so we can talk through the requirement specifications (done by Brenda Lin)	

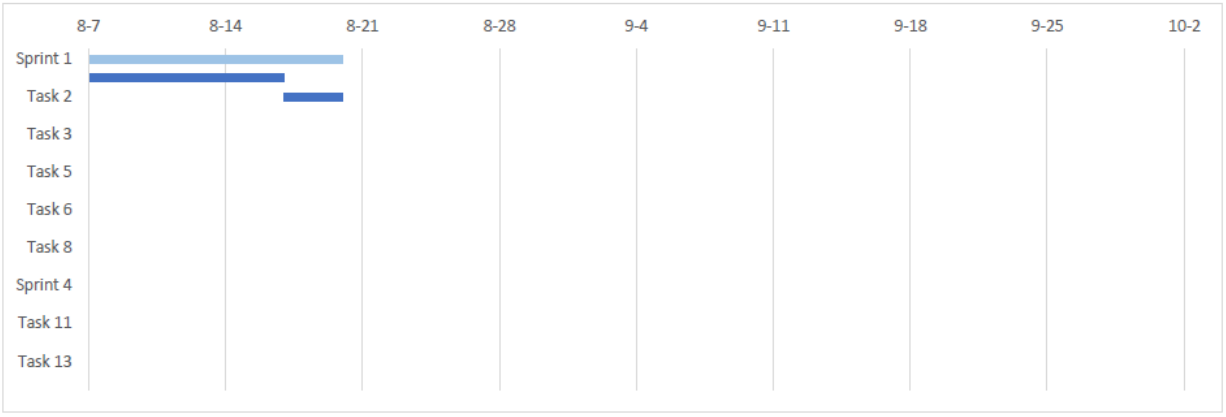
BOARD - 20 AUGUST

TO DO	IN PROGRESS	DONE	VERFICATION
			Task 1 - A brief understanding of the project (verified by Jesse Wright)
			Task 2 - A meeting with the client so we can talk through the requirement specifications (verified by Liam Leibrandt)

SPRINT 1 BACKLOG

Start Date 07-Aug
End Date 02-Oct

Task	Start Date	End Date	Duration (Days)	Status	Priority
Sprint 1	8-7	8-20	13	Complete	Complete
Task 1	8-7	8-17	10	Complete	High
Task 2	8-17	8-20	3	Complete	High
Sprint 2	8-21	9-3	13	Not started	Not started
Task 3	8-21	8-28	7	Not started	High
Task 4	8-29	9-1	3	Not started	High
Task 5	9-1	9-3	2	Not started	High
Sprint 3	9-4	9-17	13	Not started	Not started
Task 6	9-4	9-7	3	Not started	High
Task 7	9-5	9-8	3	Not started	High
Task 8	9-9	9-17	8	Not started	High
Task 9	9-13	9-17	4	Not started	High
Sprint 4	9-18	10-2	14	Not started	Not started
Task 10	9-18	9-22	4	Not started	High
Task 11	9-20	9-24	4	Not started	High
Task 12	9-23	9-27	4	Not started	High
Task 13	9-28	10-2	4	Not started	High
Task 14	9-30	10-2	2	Not started	Low



SPRINT 2 - DATE (21 AUGUST - 3 SEPTEMBER)

BOARD - 21 AUGUST

TO DO	IN PROGRESS	DONE	VERFICATION
Task 3 -To define project specifications			
Task 4 - To create database			
Task 5 - To create the user interface			

BOARD - 28 AUGUST

TO DO	IN PROGRESS	DONE	VERFICATION
		Task 3 -To define project specifications (done by Brenda Lin)	
	Task 4 - To create database (shifted by Jesse Wright)		
Task 5 - To create the user interface			

BOARD - 1 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
			Task 3 -To define project specifications (verified by Abdulkadir Dere)
		Task 4 - To create database (done by Liam Leibrandt)	
	Task 5 - To create the user interface (shifted by Brenda Lin)		

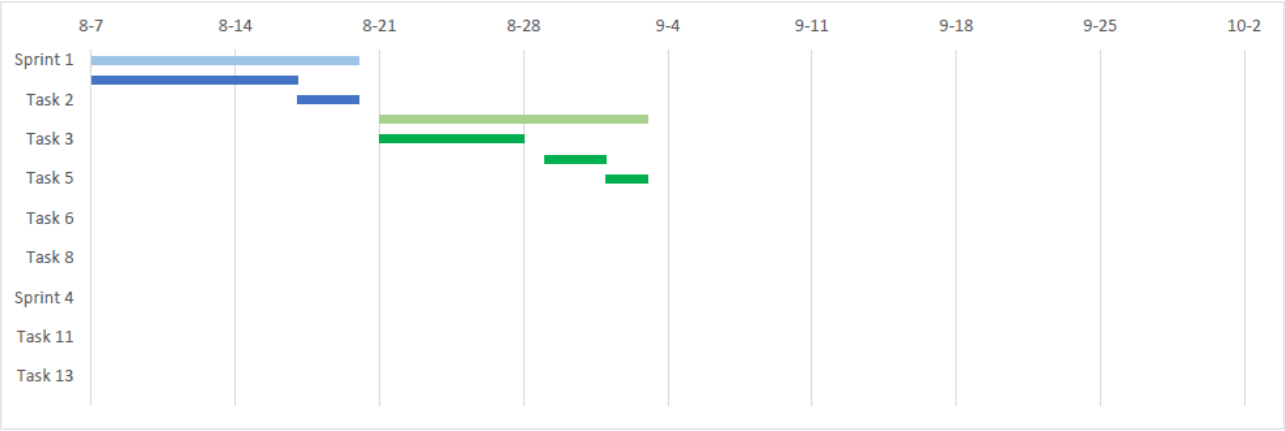
BOARD - 3 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
			Task 3 -To define project specifications (verified by Abdulkadir Dere)
			Task 4 - To create database (verified by Abdulkadir Dere)
			Task 5 - To create the user interface (done by Jesse Wright, verified by Brenda Lin)

SPRINT 2 BACKLOG

Start Date 07-Aug
End Date 02-Oct

Task	Start Date	End Date	Duration (Days)	Status	Priority
Sprint 1	8-7	8-20	13	Complete	
Task 1	8-7	8-17	10	Complete	High
Task 2	8-17	8-20	3	Complete	High
Sprint 2	8-21	9-3	13	Complete	
Task 3	8-21	8-28	7	Complete	High
Task 4	8-29	9-1	3	Complete	High
Task 5	9-1	9-3	2	Complete	High
Sprint 3	9-4	9-17	13	Not started	
Task 6	9-4	9-7	3	Not started	High
Task 7	9-5	9-8	3	Not started	High
Task 8	9-9	9-17	8	Not started	High
Task 9	9-13	9-17	4	Not started	High
Sprint 4	9-18	10-2	14	Not started	
Task 10	9-18	9-22	4	Not started	High
Task 11	9-20	9-24	4	Not started	High
Task 12	9-23	9-27	4	Not started	High
Task 13	9-28	10-2	4	Not started	High
Task 14	9-30	10-2	2	Not started	Low



SPRINT 3 - DATE (4 SEPTEMBER - 17 SEPTEMBER)

BOARD - 4 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
Task 6 -The user to be able to login			
Task 7 - Applicant to be able to create an			
Task 8 - To save documents to the database			
Task 9 - View applications			

BOARD - 5 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
	Task 6 -The user to be able to login (shifted by Liam Leibrandt)		
	Task 7 - Applicant to be able to create an (shifted by Jesse Wright)		
Task 8 - To save documents to the database			
Task 9 - View applications			

BOARD - 7 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
		Task 6 -The user to be able to login (done by Abdulkadir Dere)	
	Task 7 - Applicant to be able to create an (shifted by Jesse Wright)		
Task 8 - To save documents to the database			
Task 9 - View applications			

BOARD - 8 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
			Task 6 -The user to be able to login (verified by Brenda Lin)
		Task 7 - Applicant to be able to create an (done by Liam Leibrandt)	
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
Task 9 - View applications			

BOARD - 13 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
			Task 6 -The user to be able to login (verified by Brenda Lin)
			Task 7 - Applicant to be able to create an (verified by Jesse Wright)
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
	Task 9 - View applications (shifted by Brenda Lin)		

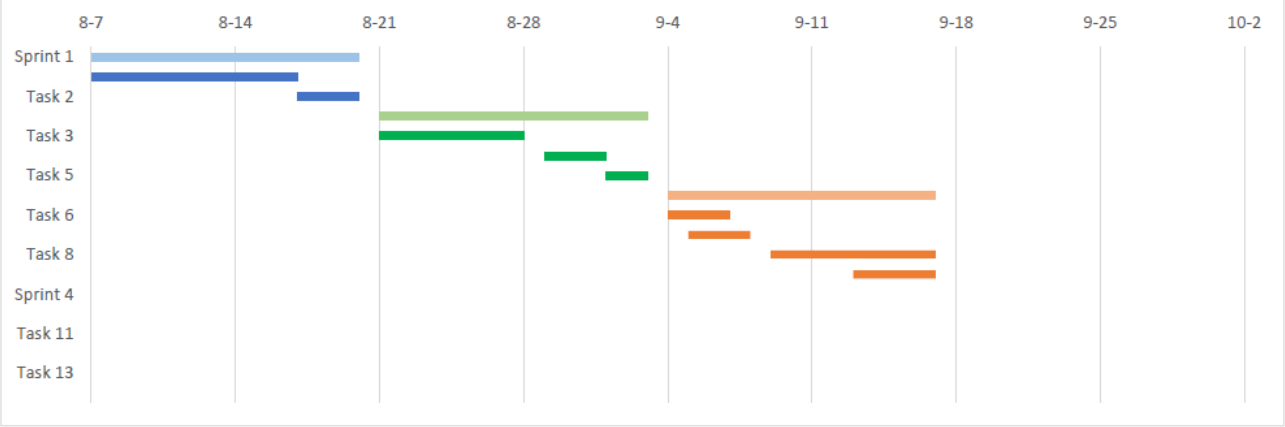
BOARD - 17 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
			Task 6 -The user to be able to login (verified by Brenda Lin)
			Task 7 - Applicant to be able to create an (verified by Jesse Wright)
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
			Task 9 - View applications (done by Liam Leibrandt, verified by Abdulkadir Dere)

SPRINT 3 BACKLOG

Start Date 07-Aug
End Date 02-Oct

Task	Start Date	End Date	Duration (Days)	Status	Priority
Sprint 1	8-7	8-20	13	Complete	
Task 1	8-7	8-17	10	Complete	High
Task 2	8-17	8-20	3	Complete	High
Sprint 2	8-21	9-3	13	Complete	
Task 3	8-21	8-28	7	Complete	High
Task 4	8-29	9-1	3	Complete	High
Task 5	9-1	9-3	2	Complete	High
Sprint 3	9-4	9-17	13	Not Complete	
Task 6	9-4	9-7	3	Complete	High
Task 7	9-5	9-8	3	Complete	High
Task 8	9-9	9-17	8	In Progress	High
Task 9	9-13	9-17	4	Complete	High
Sprint 4	9-18	10-2	14	Not started	
Task 10	9-18	9-22	4	Not started	High
Task 11	9-20	9-24	4	Not started	High
Task 12	9-23	9-27	4	Not started	High
Task 13	9-28	10-2	4	Not started	High
Task 14	9-30	10-2	2	Not started	Low



SPRINT 4 - DATE (18 SEPTEMBER - 2 OCTOBER)

BOARD - 18 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
Task 10 - Request an interview			
Task 11 - Recommend application			
Task 12 - Accept/decline application			
Task 13 -Email's to be sent at the relevant			
Task 14 - The system to have the option of printing documents			

BOARD - 20 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
	Task 10 - Request an interview (shifted by Brenda Lin)		
	Task 11 - Recommend application (shifted by Jesse Wright)		
Task 12 - Accept/decline application			
Task 13 -Email's to be sent at the relevant			
Task 14 - The system to have the option of printing documents			

BOARD - 22 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
		Task 10 - Request an interview (done by Liam Leibrandt)	
	Task 11 - Recommend application (shifted by Jesse Wright)		
	Task 12 - Accept/decline application (shifted by Liam Leibrandt)		
Task 13 -Email's to be sent at the relevant			
Task 14 - The system to have the option of printing documents			

BOARD - 24 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
			Task 10 - Request an interview (verified by Jesse Wright)
		Task 11 - Recommend application (done by Brenda Lin)	
	Task 12 - Accept/decline application (shifted by Liam Leibrandt)		
Task 13 -Email's to be sent at the relevant			
Task 14 - The system to have the option of printing documents			

BOARD - 27 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
			Task 10 - Request an interview (verified by Jesse Wright)
			Task 11 - Recommend application (verified by Liam Leibrandt)
		Task 12 - Accept/reject application (done by Jesse Wright)	
	Task 13 -Email's to be sent at the relevant (SHIFTED BY)		
Task 14 - The system to have the option of printing documents			

BOARD - 30 SEPTEMBER

TO DO	IN PROGRESS	DONE	VERFICATION
	Task 8 - To save documents to the database (shifted by Abdulkadir Dere)		
			Task 10 - Request an interview (verified by Jesse Wright)
			Task 11 - Recommend application (verified by Liam Leibrandt)
			Task 12 - Accept/decline application (verified by Brenda Lin)
	Task 13 -Email's to be sent at the relevant (shifted by Liam Leibrandt)		
	Task 14 - The system to have the option of printing documents (shifted by Abdulkadir Dere)		

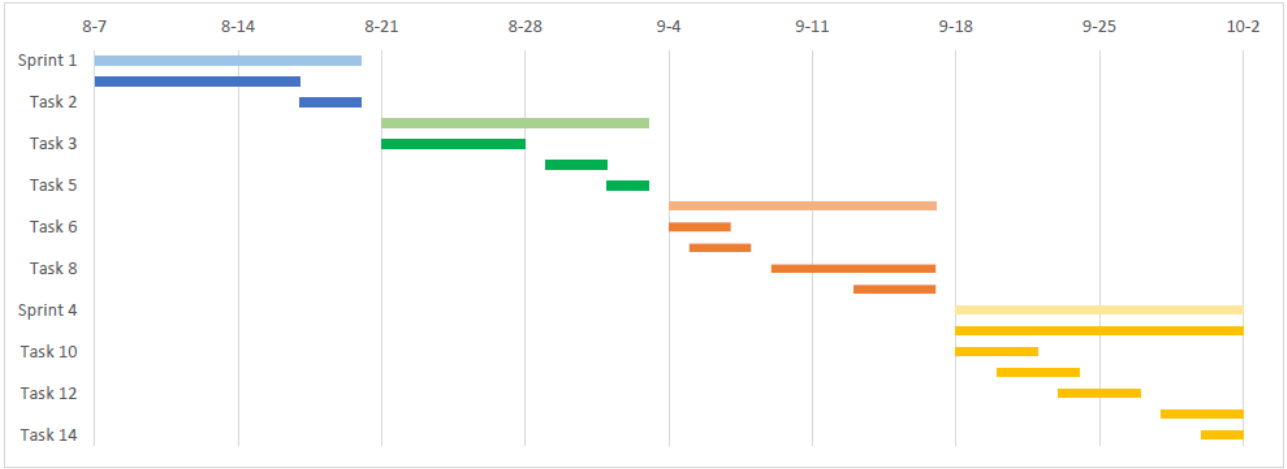
BOARD - 2 OCTOBER

TO DO	IN PROGRESS (NB: NOT COMPLETE)	DONE	VERFICATION
			Task 8 - To save documents to the database (done by Liam Leibrandt, verified by Brenda Lin)
			Task 10 - Request an interview (verified by Jesse Wright)
			Task 11 - Recommend application (verified by Liam Leibrandt)
			Task 12 - Accept/decline application (verified by Brenda Lin)
			Task 13 -Email's to be sent at the relevant (done by Abdulkadir Dere, verified by Jesse Wright)
	Task 14 - The system to have the option of printing documents (shifted by Abdulkadir Dere)		

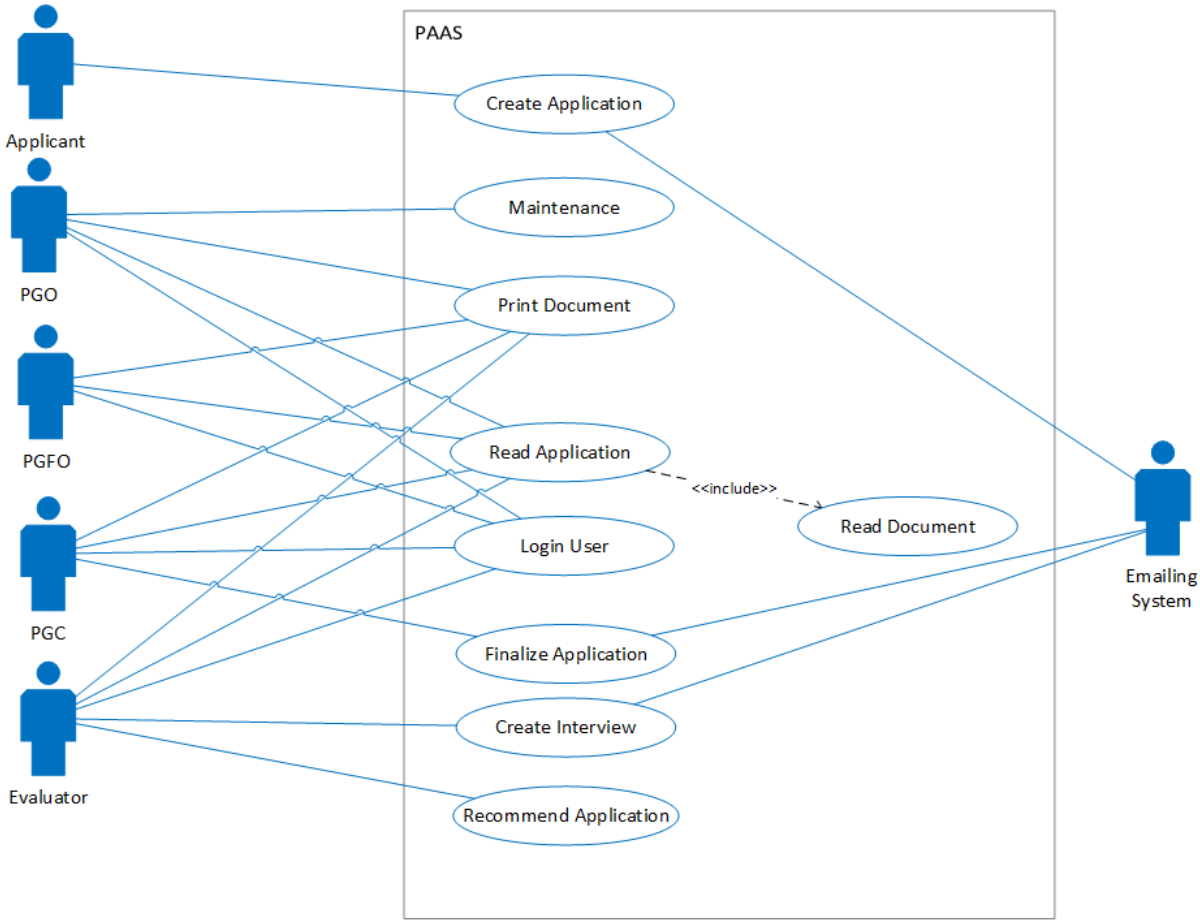
SPRINT 4 BACKLOG

Start Date 07-Aug
End Date 02-Oct

Task	Start Date	End Date	Duration (Days)	Status	Priority
Sprint 1	8-7	8-20	13	Complete	
Task 1	8-7	8-17	10	Complete	High
Task 2	8-17	8-20	3	Complete	High
Sprint 2	8-21	9-3	13	Complete	
Task 3	8-21	8-28	7	Complete	High
Task 4	8-29	9-1	3	Complete	High
Task 5	9-1	9-3	2	Complete	High
Sprint 3	9-4	9-17	13	Complete	
Task 6	9-4	9-7	3	Complete	High
Task 7	9-5	9-8	3	Complete	High
Task 8	9-9	9-17	8	Complete	High
Task 9	9-13	9-17	4	Complete	High
Sprint 4	9-18	10-2	14	Not Complete	
Task 8	9-18	10-2	14	Complete	High
Task 10	9-18	9-22	4	Complete	High
Task 11	9-20	9-24	4	Complete	High
Task 12	9-23	9-27	4	Complete	High
Task 13	9-28	10-2	4	Complete	High
Task 14	9-30	10-2	2	Not Complete	Low



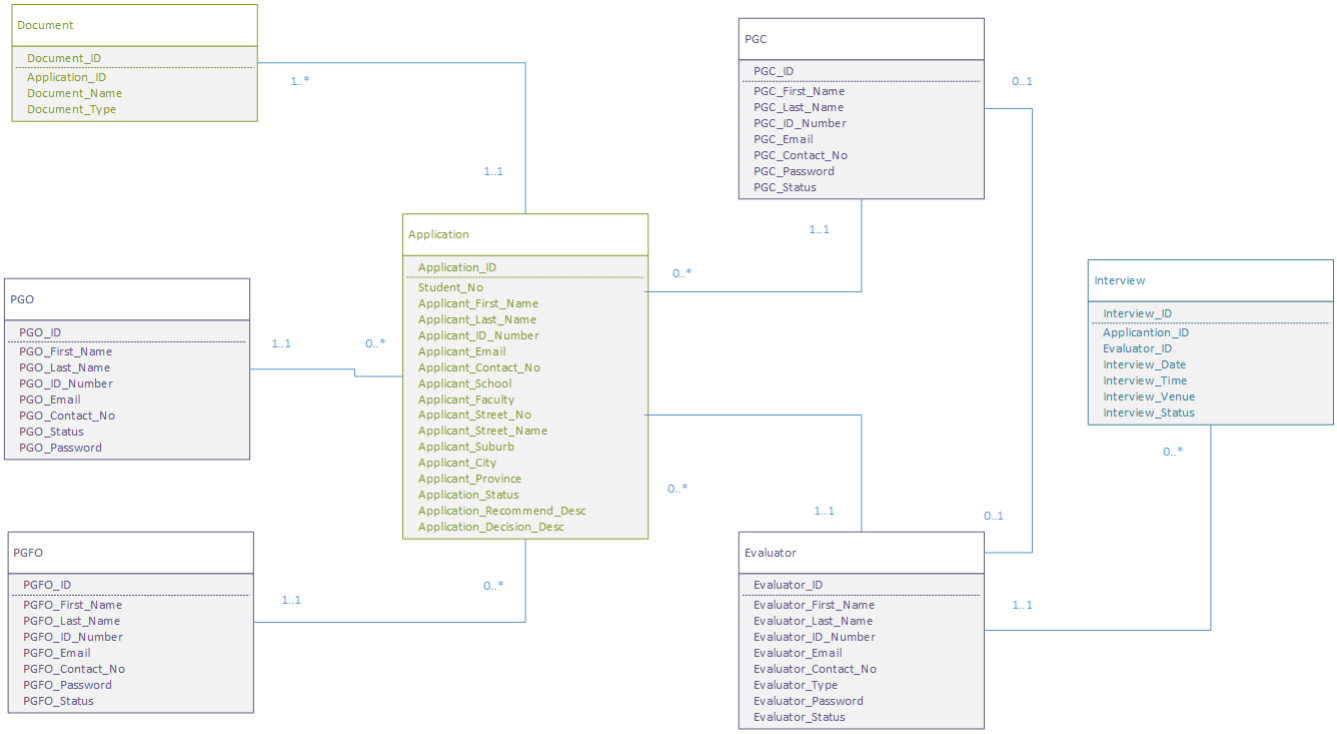
3.9 Use Case Diagram



Use Case Diagram for PAAS

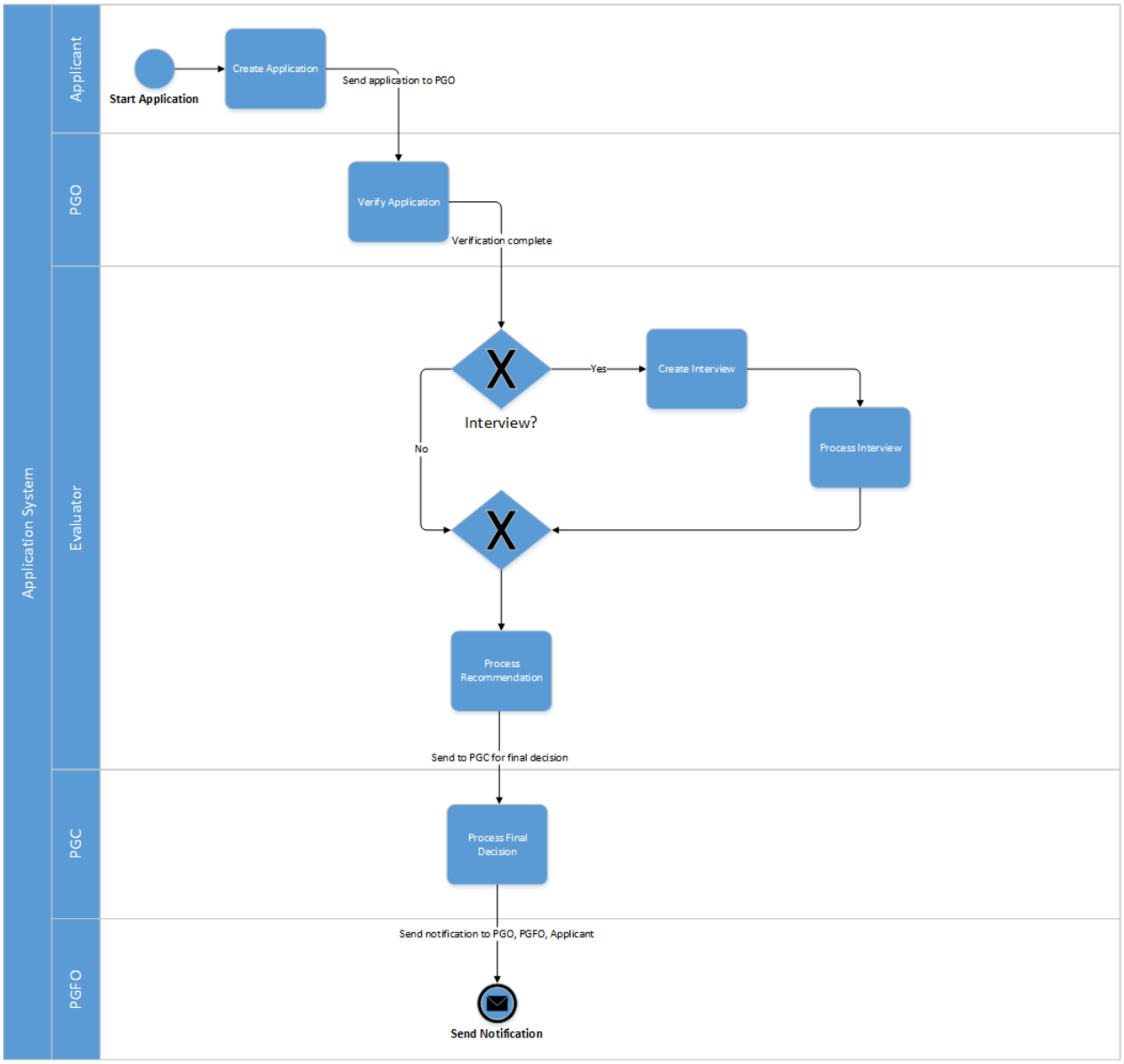
Note: Maintenance use case in the diagram refers to all the maintenance use cases specified in section 2.2.2

3.10 Class Model Diagram

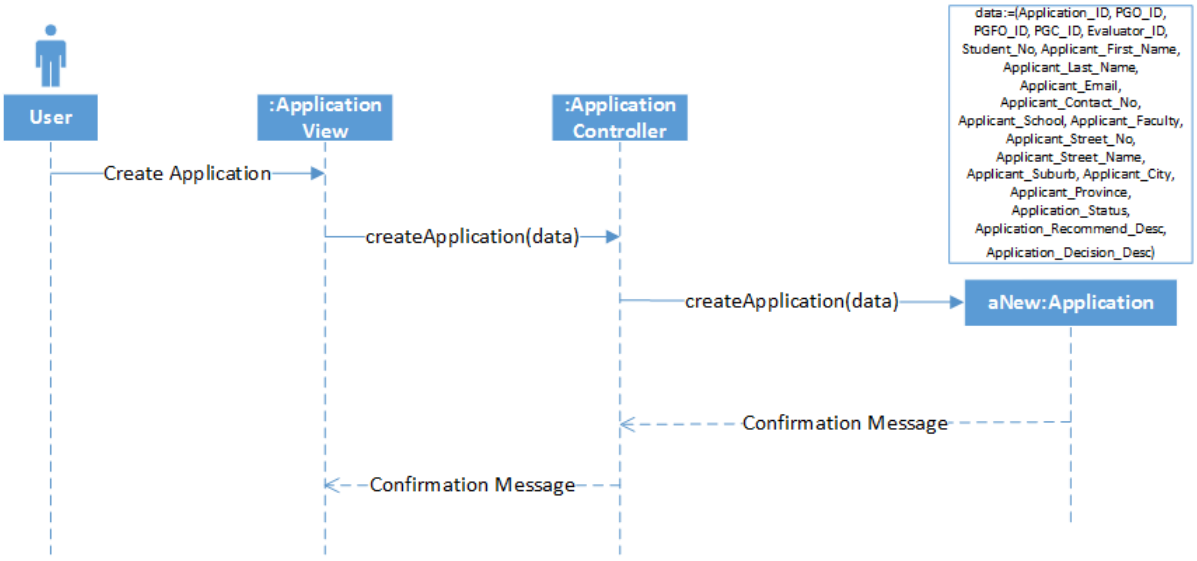


Class Diagram Model: PAAS System

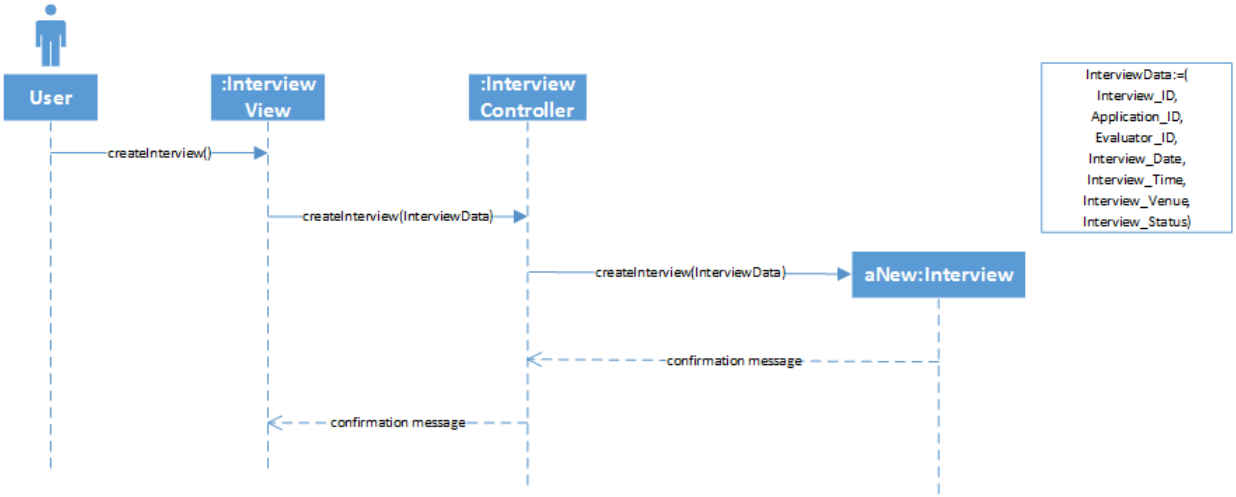
3.11 Process Model (Flow Model)



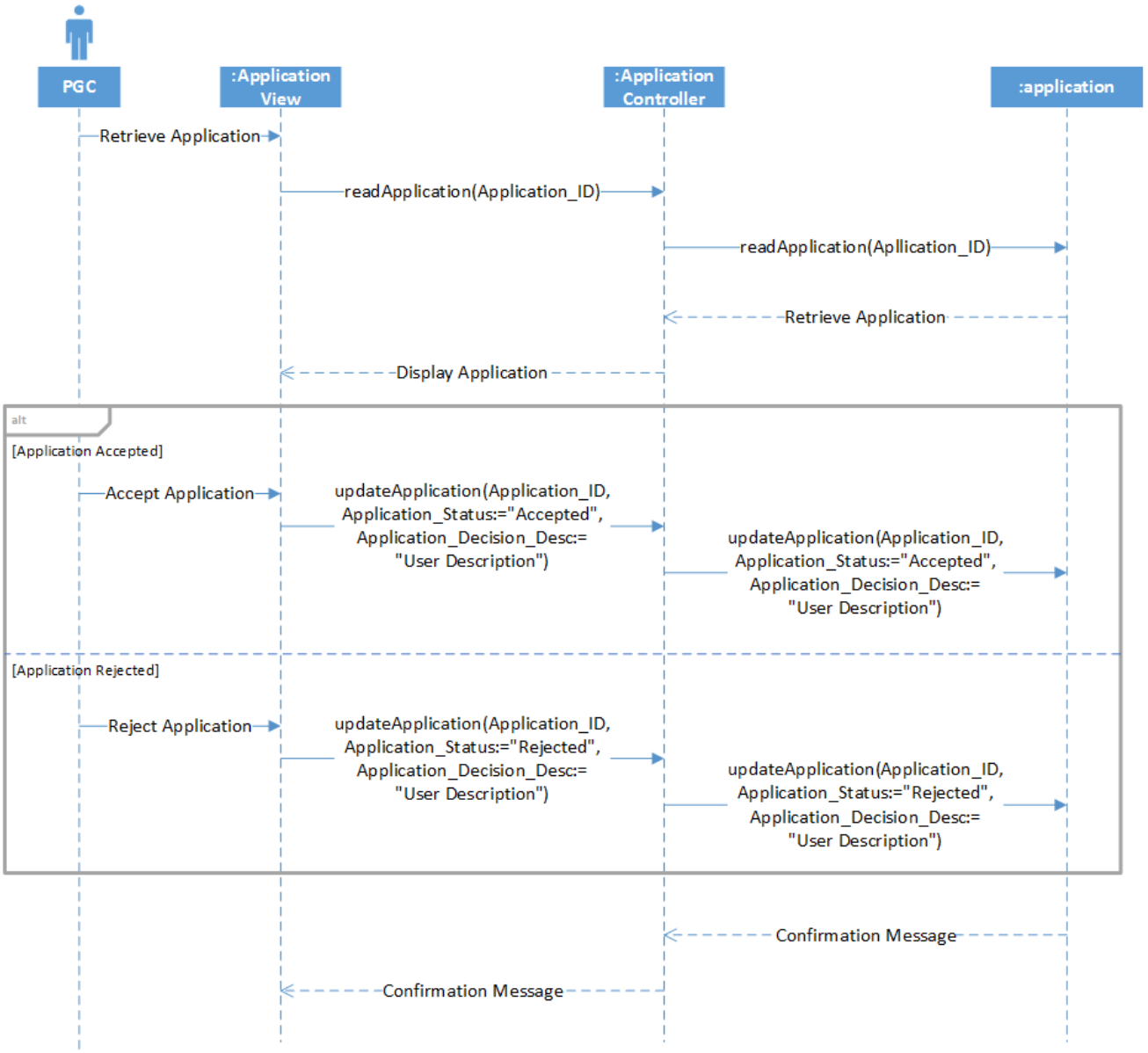
3.12 Sequence Diagrams



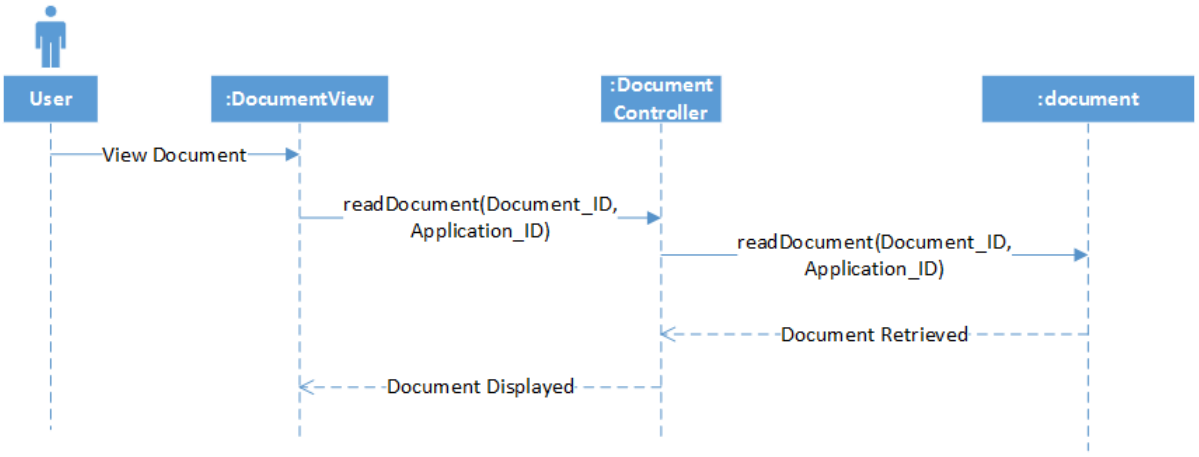
Sequence Diagram: Create Application



Sequence Diagram: Create Interview

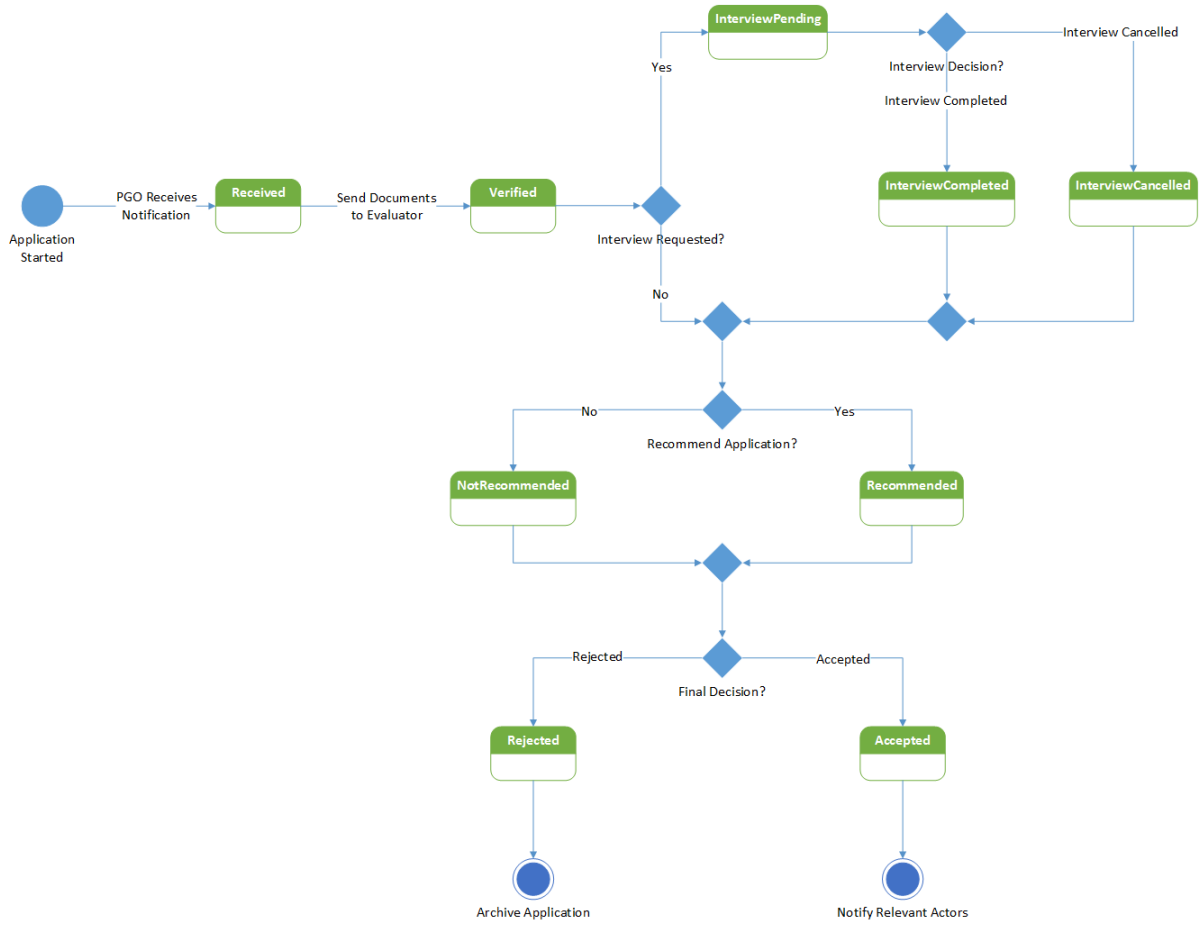


Sequence Diagram: Finalize Application

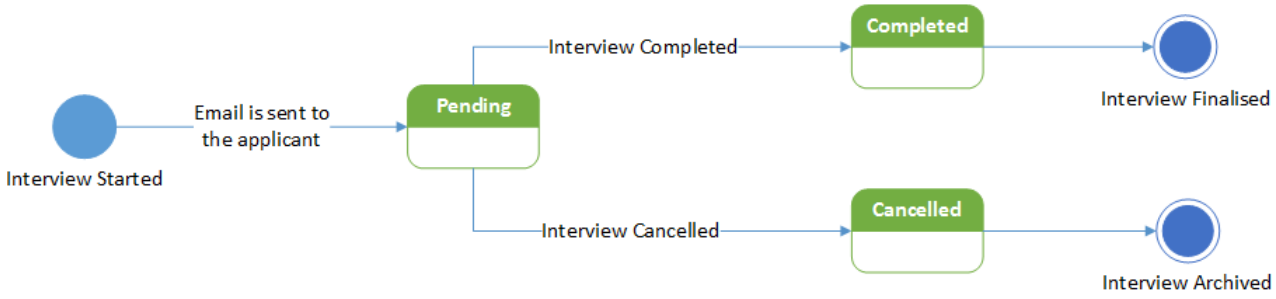


Sequence Diagram: Read Document

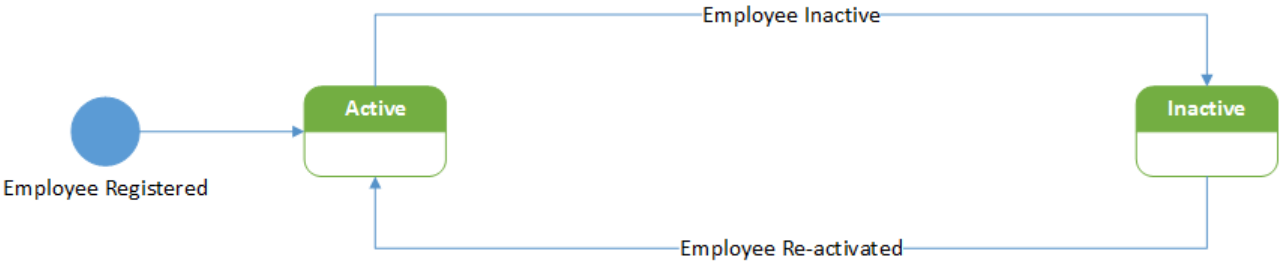
3.13 State Machine Diagrams



State Machine Diagram: Application Status Object



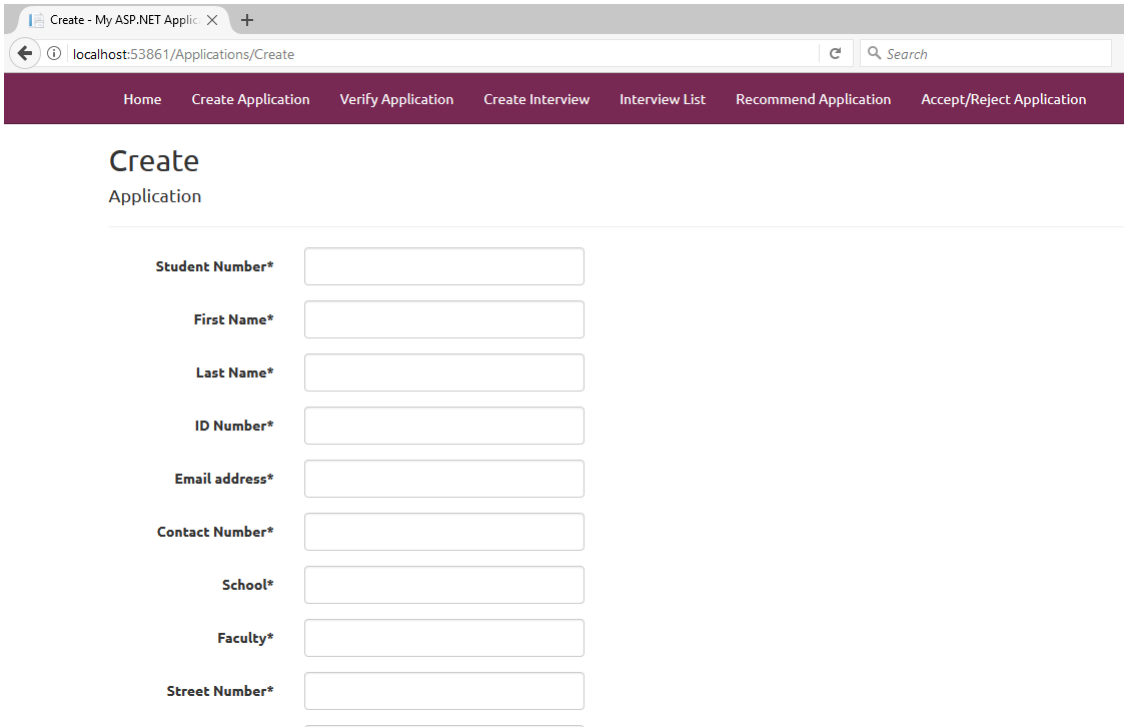
State Machine Diagram: Interview Status Object

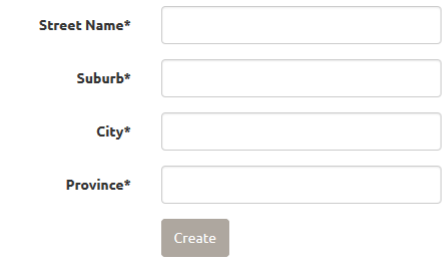


State Machine Diagram: Employee Status Object

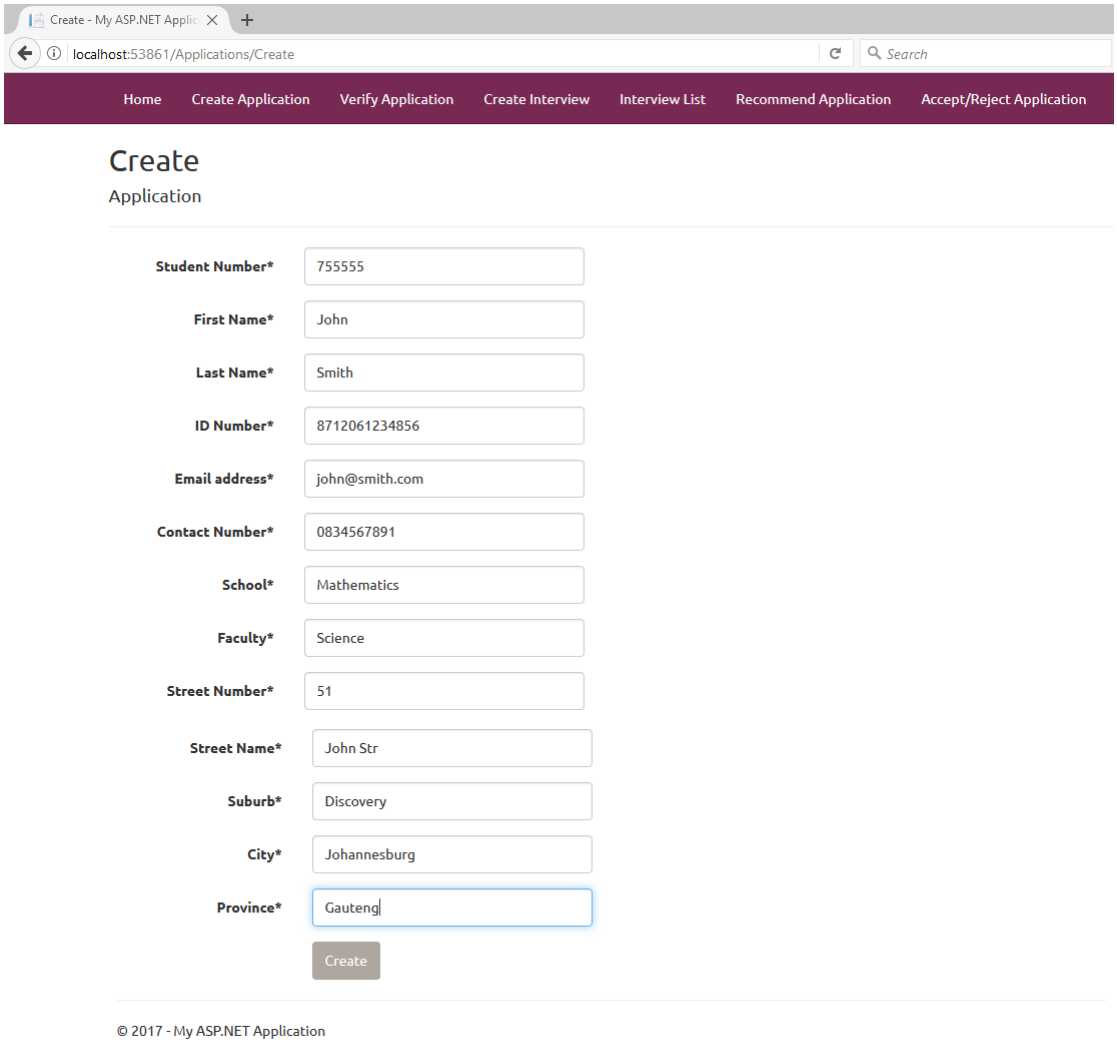
4 Implementation (User Manual)

4.1 Create Application

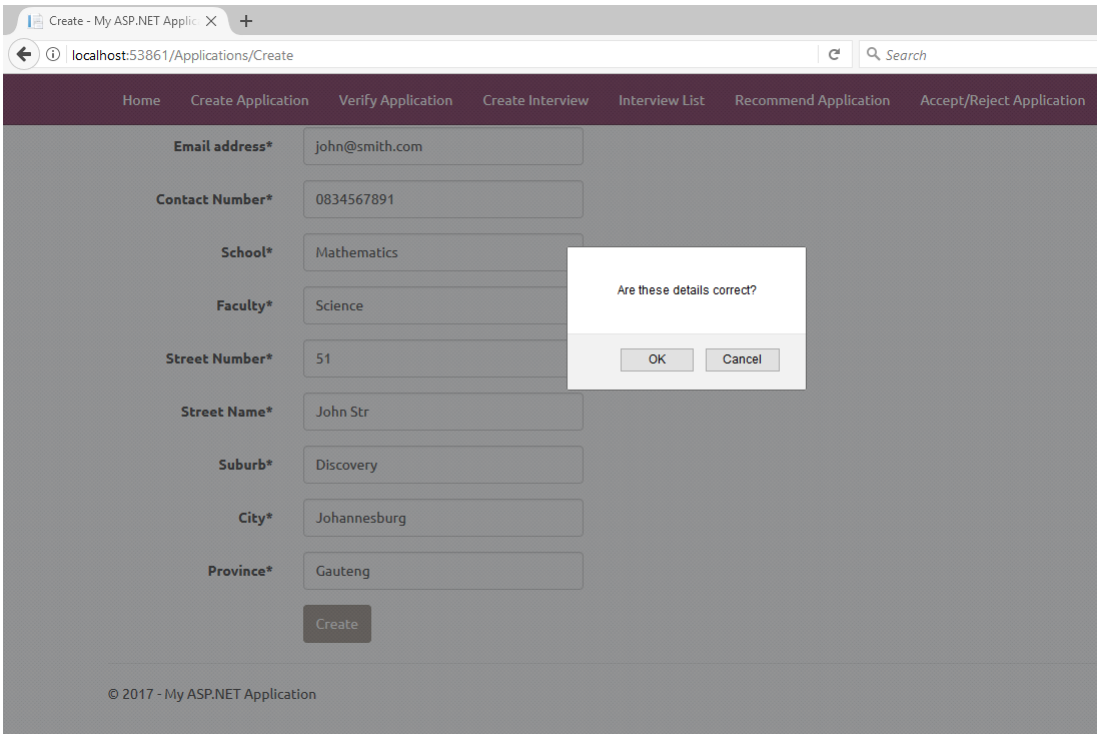




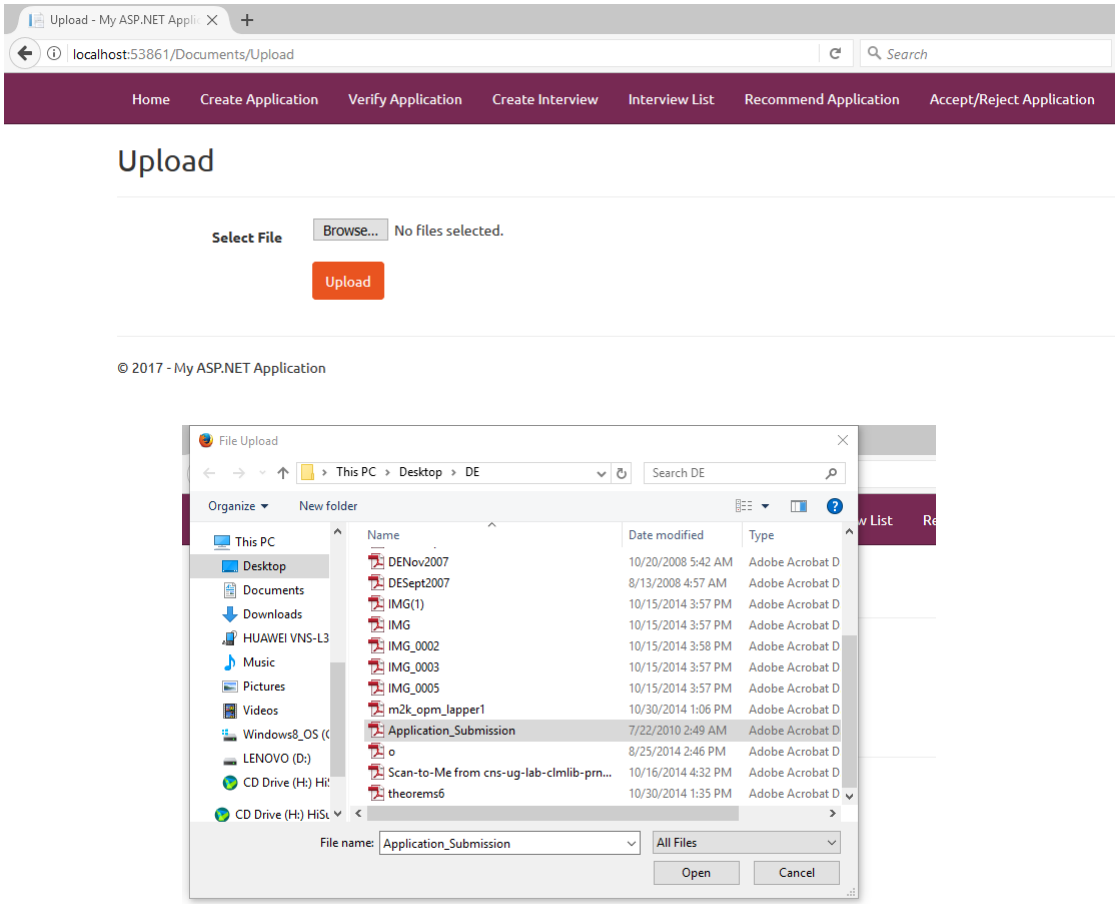
This is the application creation page.



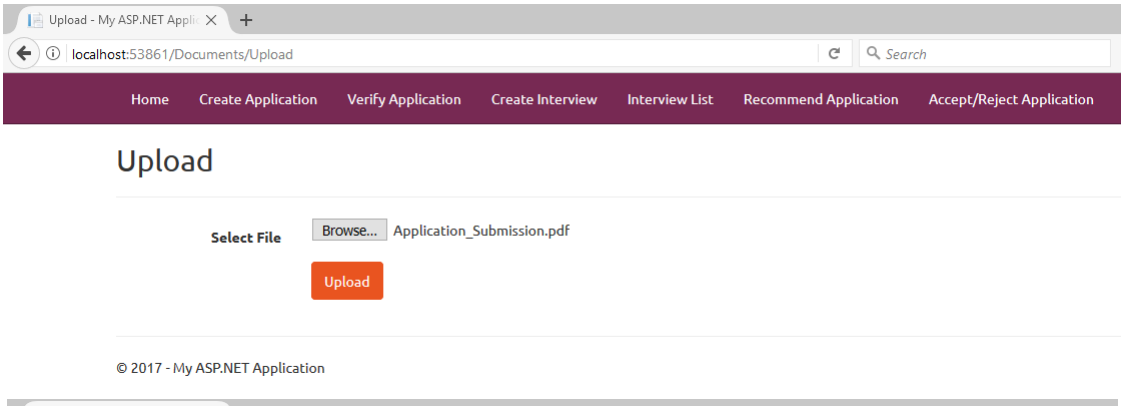
Fill in the required details and click create.

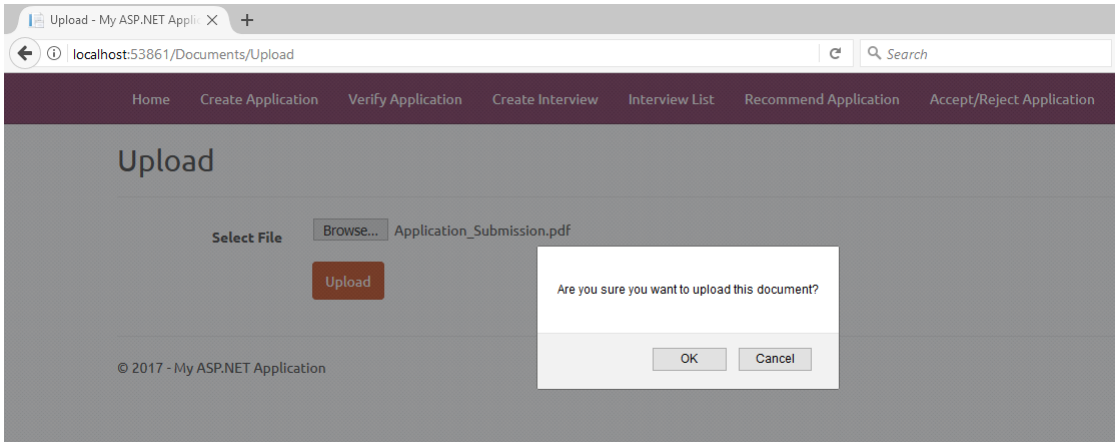


Confirm your details.



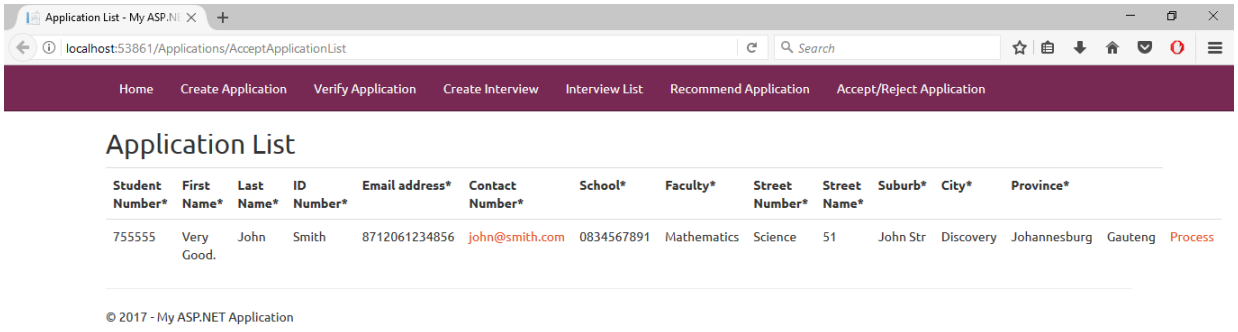
Browse for application documents to submit.



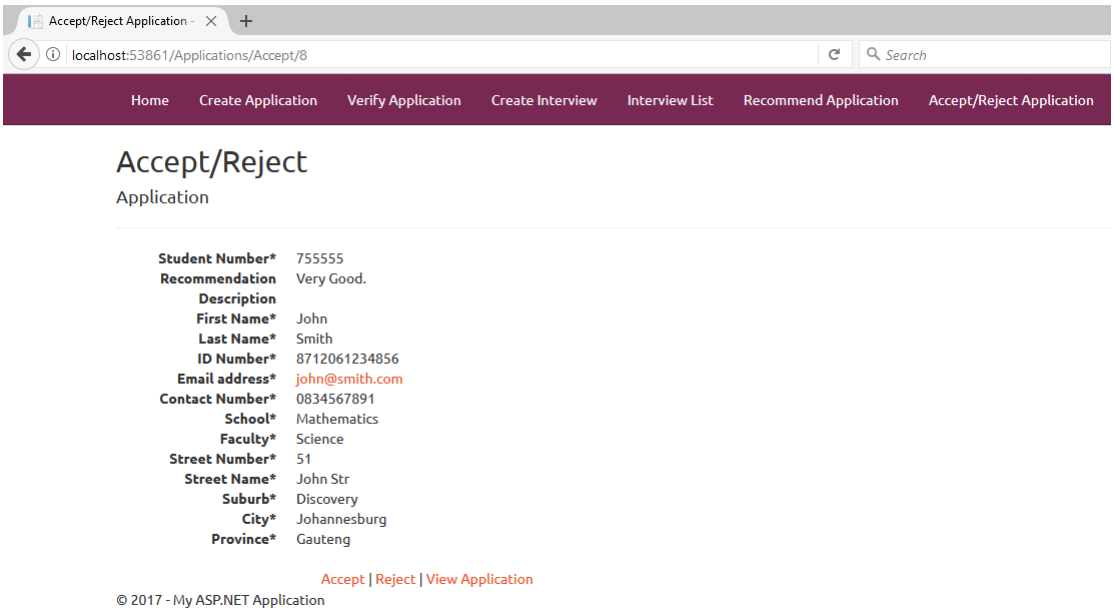


Upload the document and confirm the submission.

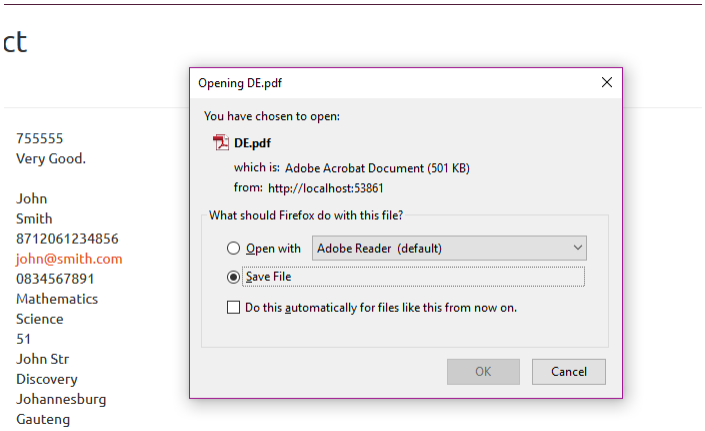
4.2 Accept Application



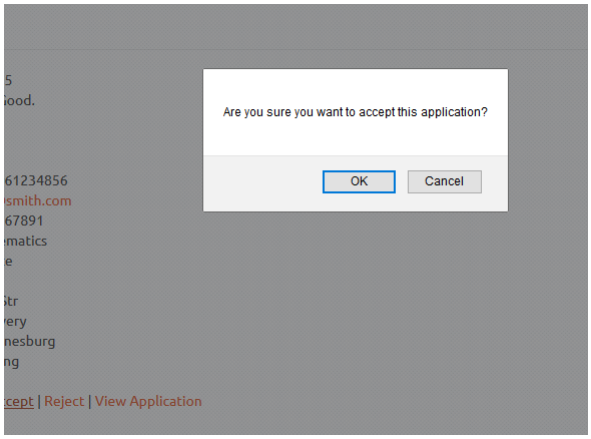
This is the application index list. Click on process.



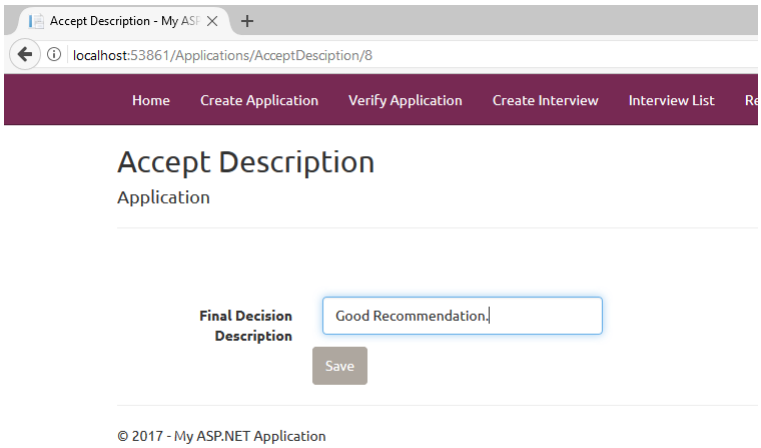
Click view application to download respective application documents.



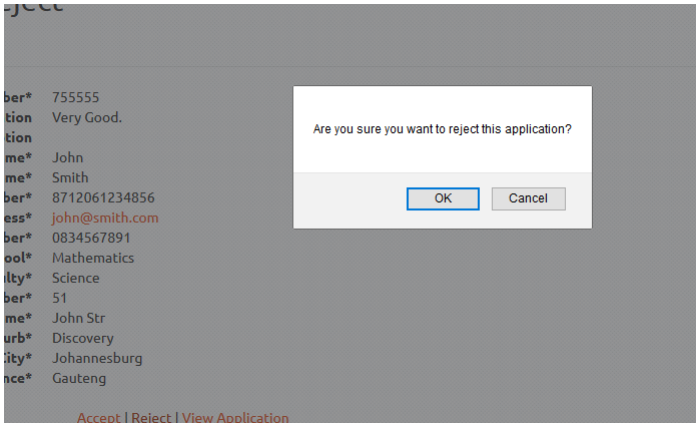
Save the document.



Click accept to accept the application and confirm.

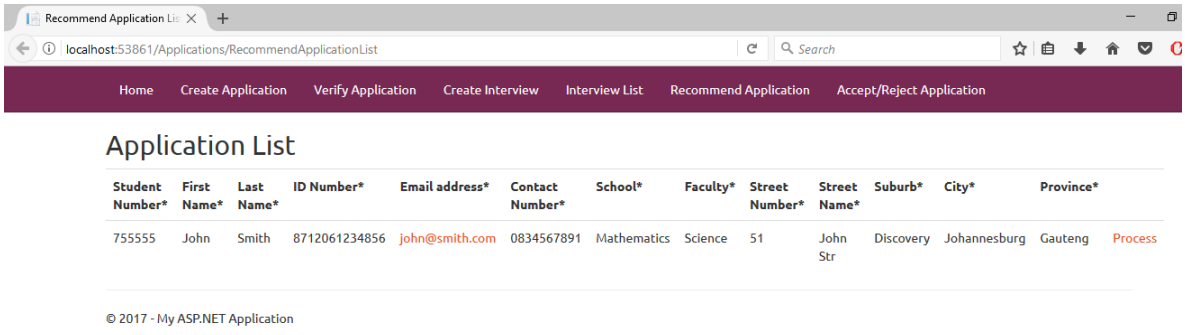


Leave feedback on the accepted application.

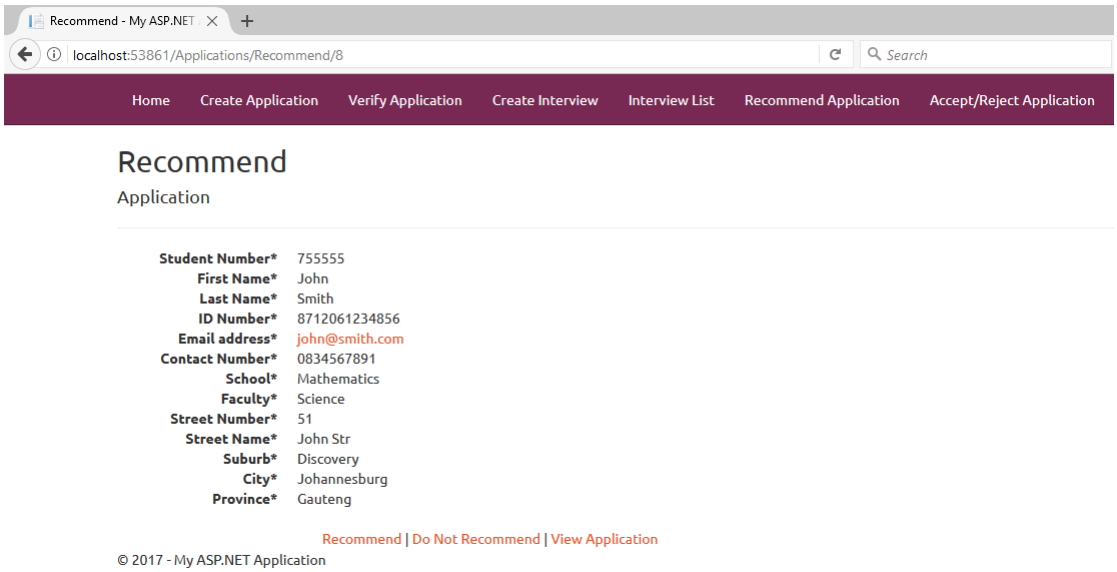


Click reject to reject the application and confirm.

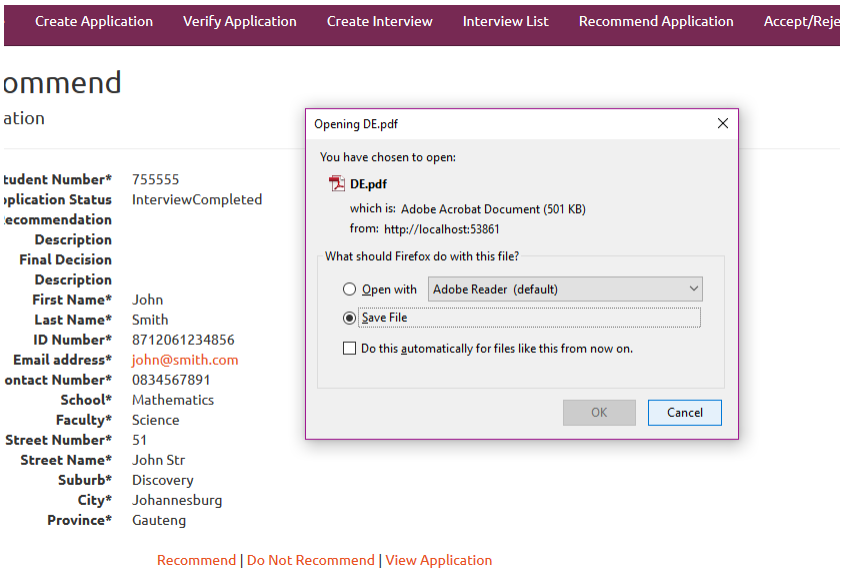
4.3 Recommend Application



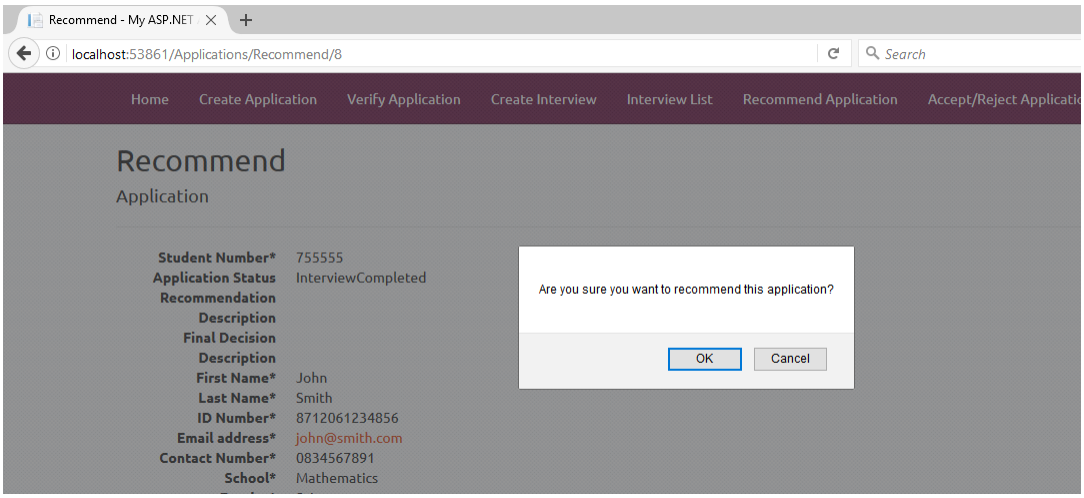
This is the application index list. Click on process.



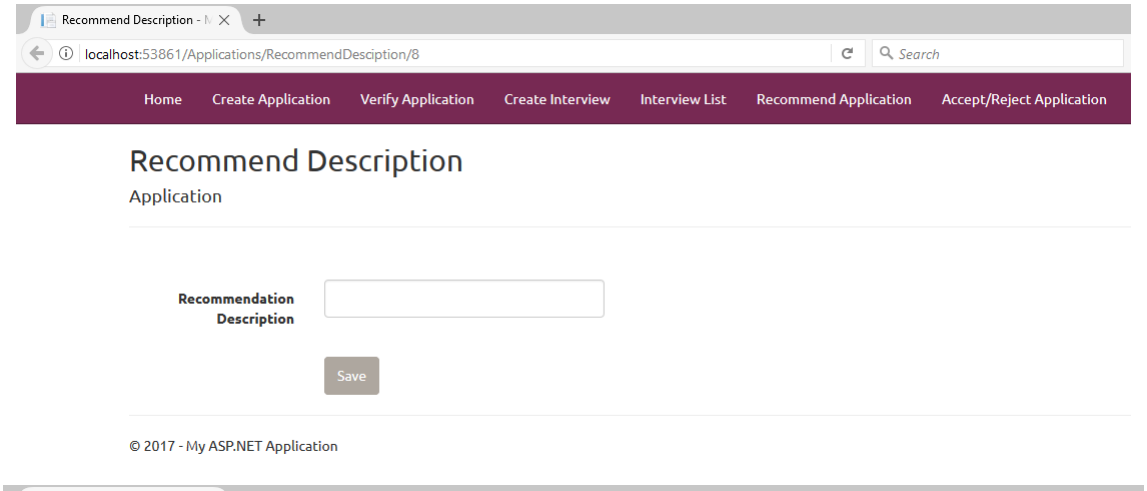
Click view application to download respective application documents.

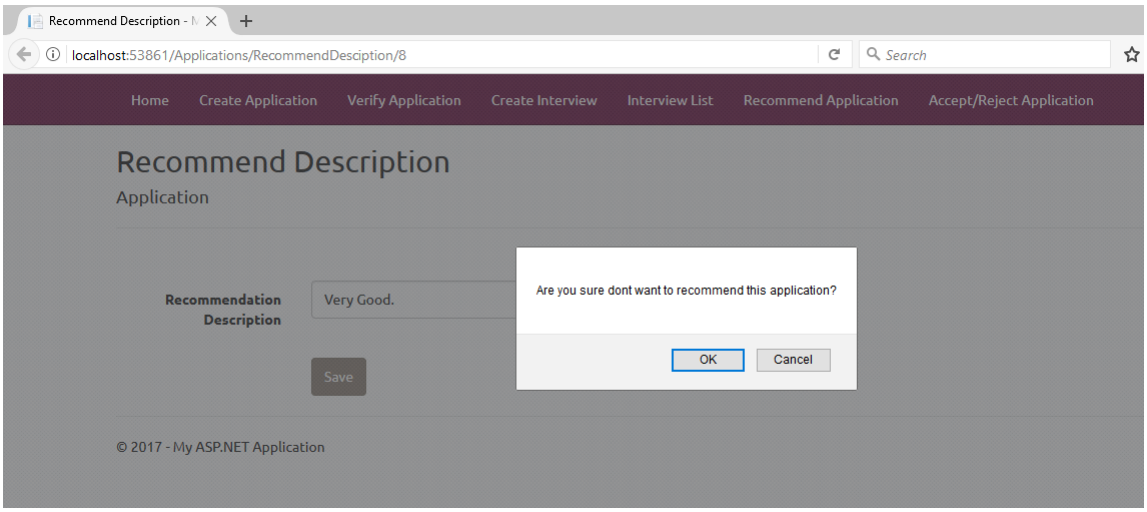


Save the document.

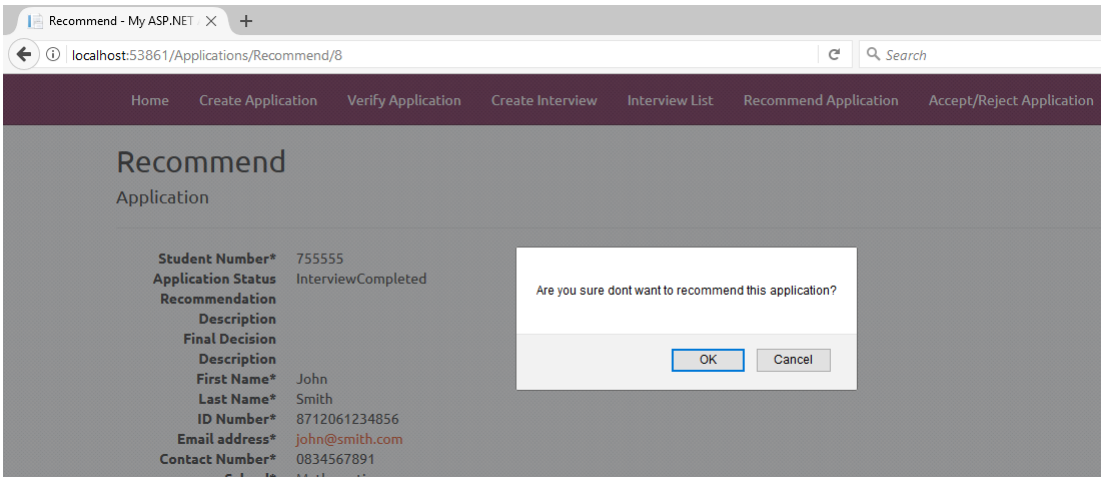


Click recommend to recommend the application and confirm.



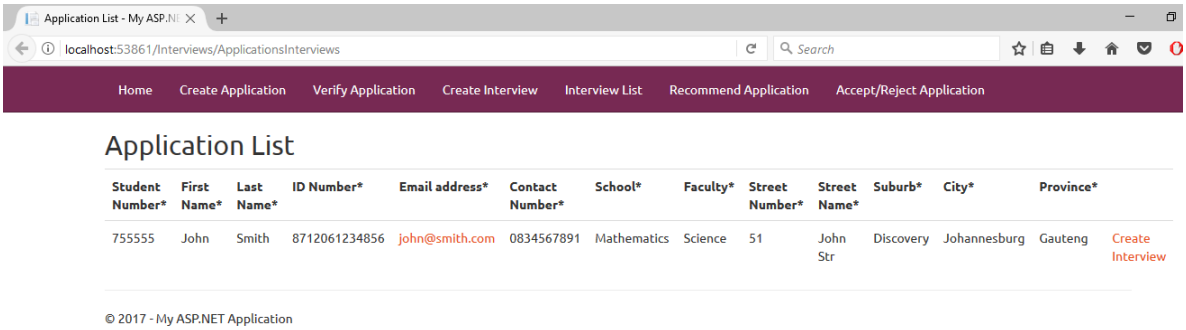


Leave feedback on the recommendation and confirm.

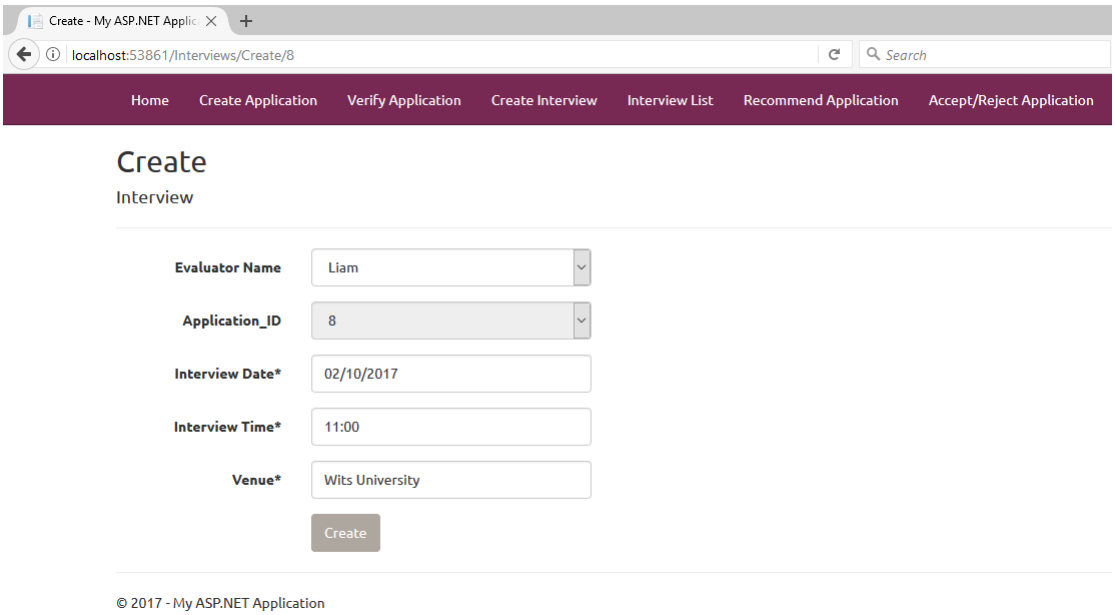


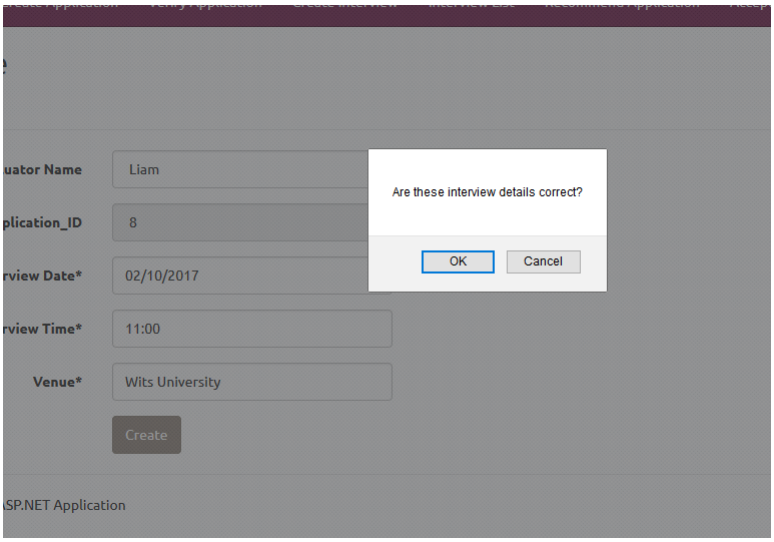
Click do not recommend to not recommend the application and confirm.

4.4 Create Interview



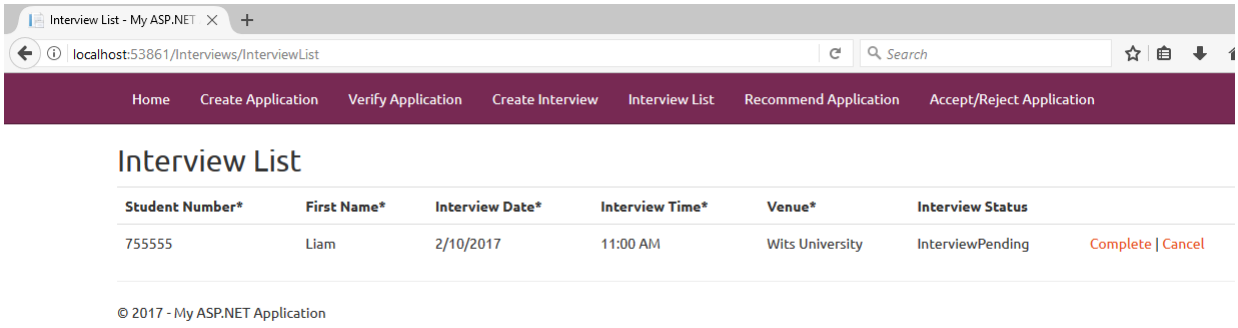
This is the application index list. Click on create interview.



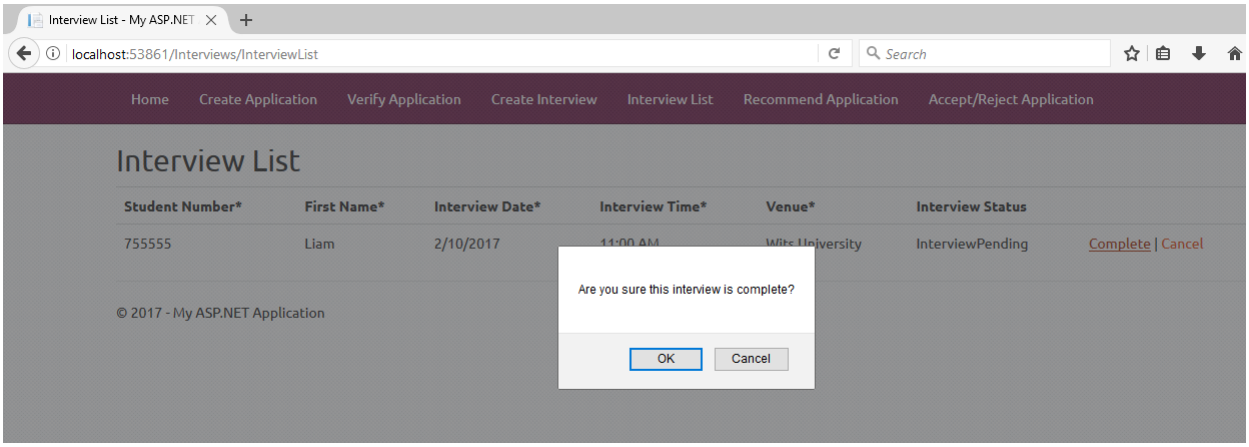


Fill in the appropriate details and confirm the interview slot.

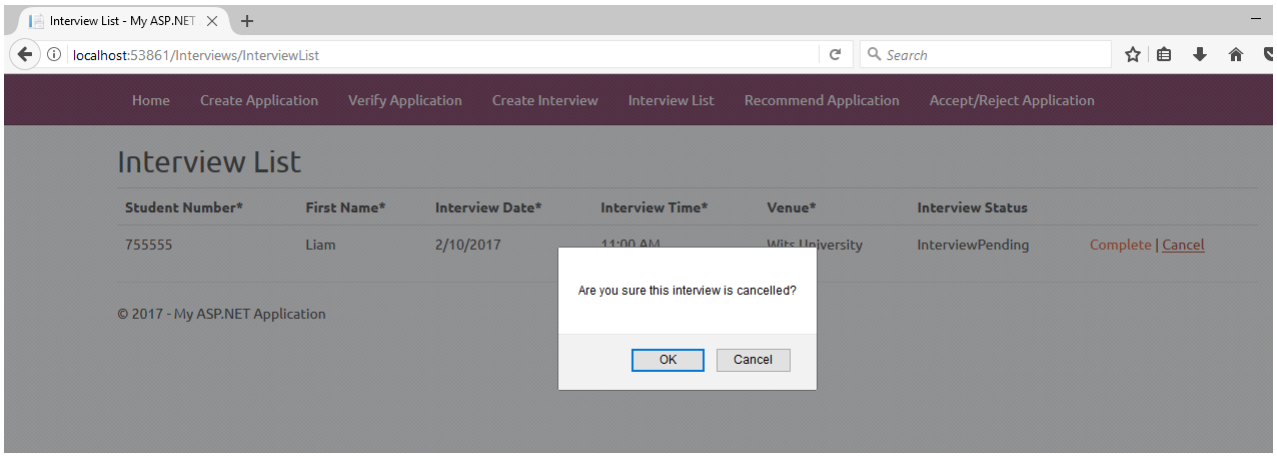
4.5 Process Interview



This is the interview index list. Click on complete.

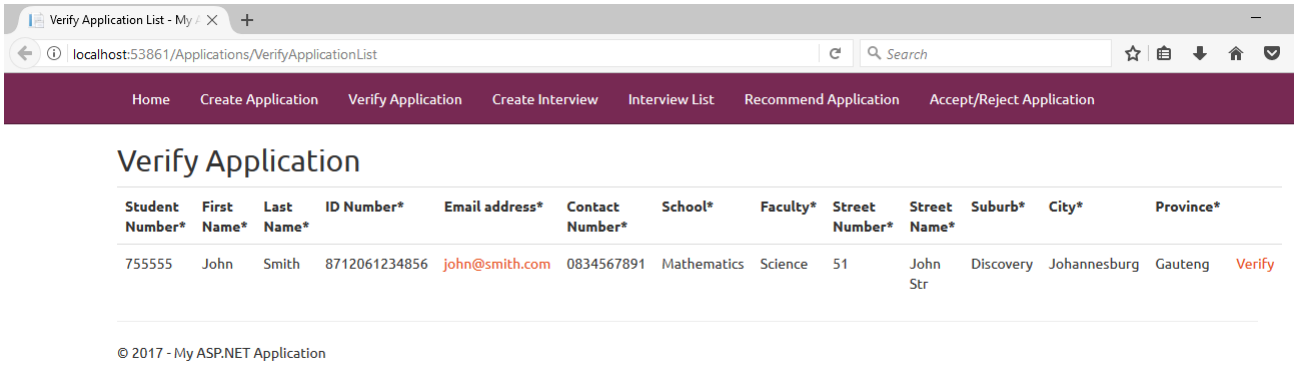


Confirm that the interview has been completed.

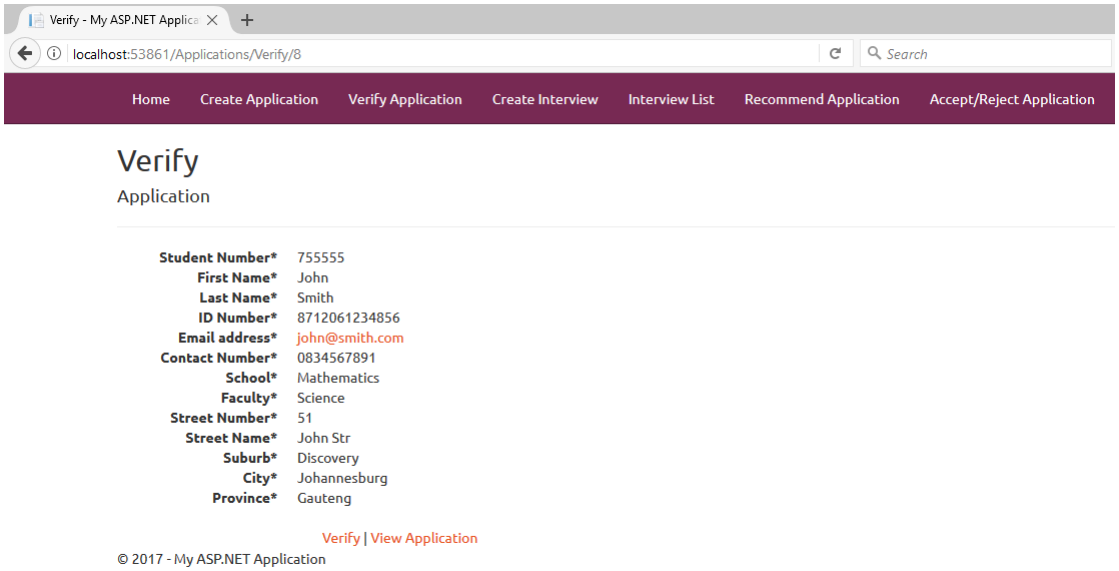


Click on cancel and confirm that the interview has been cancelled.

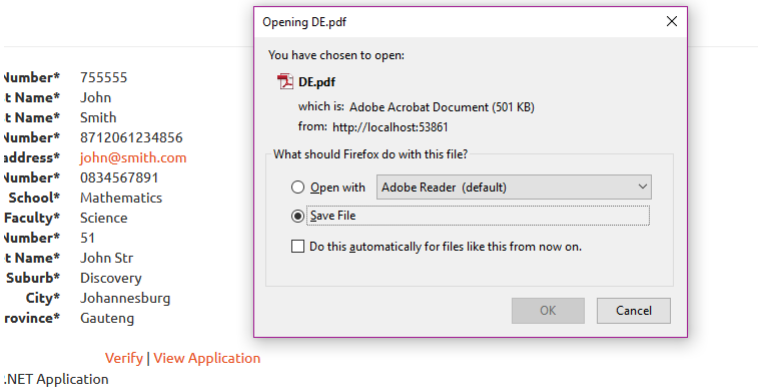
4.6 Verify Application



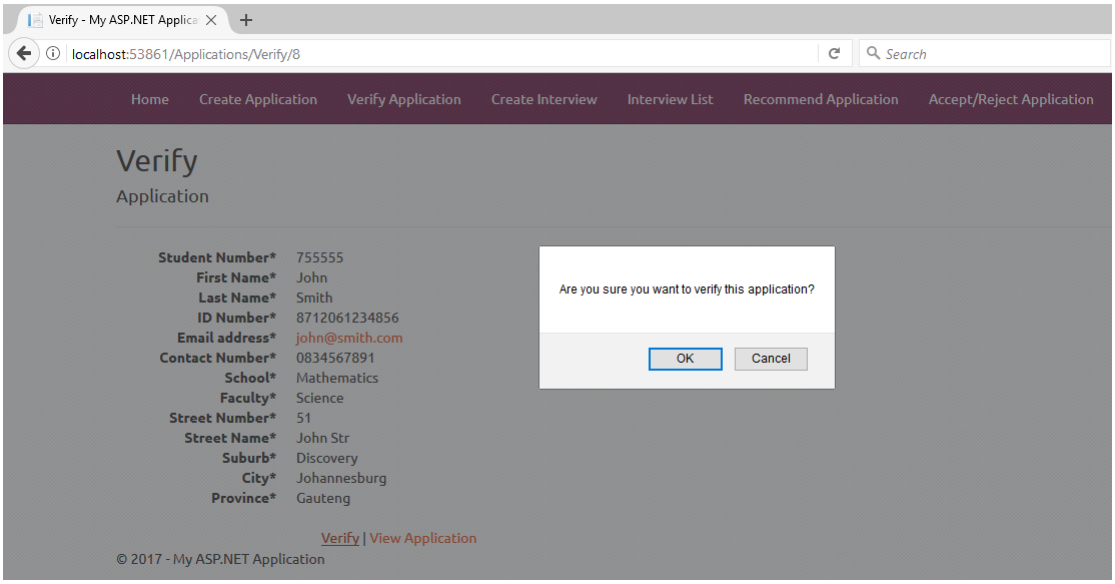
This is the application verification index list. Click verify.



Click view application to download respective application documents.



Save the document.



Click verify and confirm the verification.

5 Testing

5.1 Functionality Testing

5.1.1 Motivation for Functionality Testing

Functionality testing is very important because it allows us to check and ensure that all functions that we have implemented for our systems use cases, are running correctly. This ensures that the data passed from users through the functions (via field forms) to the database is accurate and without errors. This also allows us to determine if any use cases/processes contain faulty logic or flow so that we may review and alter them.

5.1.2 Case Name: Create Application

Requirement Description: The student should be able to create an application.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	Applicant will prompt the system to create a new application.		Application form opens.	The system did as expected.	Pass	
2	The system will prompt the user to enter student number, first and last name, ID number, email, contact number, school, faculty, street number, street name, suburb, city and province.	student number, first and last name, ID number, email, contact number, school, faculty, street number, street name, suburb, city and province	The form will be filled with applicants details.	The system did as expected.	Pass	
3	The system will prompt the user to upload their documentation in a .pdf format.	Documentation in pdf format.	The document(s) will be uploaded.	The system did as expected.	Pass	
4	The user will confirm the selection after the system has prompted the user.		The creation of the application will be confirmed.	The system did as expected.	Pass	

5.1.3 Case Name: Read Document

Requirement Description: The user should be able to download the a required document.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	The actor will select the document to be viewed.		The document will be downloaded.	The system did as expected.	Pass	

5.1.4 Case Name: Create Interview

Requirement Description: The user should be able to request an interview with the applicant.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	The actor will prompt the system for an interview form.		The system will open the interview form.	The system did as expected.	Pass	
2	The system will prompt the actor to enter date time and venue of the interview.	Date, time and venue	The form will be filled with interviews details.	The system did as expected.	Pass	
3	The user will confirm the selection after the system has prompted the user.		The creation of the interview will be confirmed.	The system did as expected.	Pass	

5.1.5 Case Name: Recommend Application

Requirement Description: The user should be able to recommend an application.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	The actor will prompt the system to recommend an application.		The system will open the recommendation form.	The system did as expected.	Pass	
2	The system will prompt the user to enter a recommendation description.	Recommendation description.	The form will be filled with the recommendation description.	The system did as expected.	Pass	
3	The user will confirm the selection after the system has prompted the user.		The recommendation will be confirmed.	The system did as expected.	Pass	

5.1.6 Case Name: Finalize Application

Requirement Description: The user should be able to reject or accept an application.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	The actor will prompt the system to finalize an application.		The system will open the finalization form.	The system did as expected.	Pass	
2	The system will prompt the user to enter a finalization description.	Finalization description.	The system will open the finalization form.	The system did as expected.	Pass	
3	The user will confirm the selection after the system has prompted the user.		The finalization of the application will be confirmed.	The system did as expected.	Pass	

5.1.7 Case Name: Login User

Requirement Description: The user should be able to download the a required document.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	The actor will prompt the system to login.		The system will open the login page.	The system did as expected.	Pass	
2	The system will prompt the user to enter in their email and password.	Email and password	The login details will be filled with the actors email and password.	The system did as expected.	Pass	
3	The user will confirm the selection after the system has prompted the user.		The user will be logged in.	The system did as expected.	Pass	

5.2 Security Testing

5.2.1 Motivation for Security Testing

Security testing is the process of testing the system to highlight the flaws and bugs within the system with regards to confidentiality, integrity, authentication, authorisation, availability and non-repudiation.
Security testing is important as it maintains the systems intended functionality. It also analysis the system for any weaknesses, technical flaws or vulnerabilities.

5.2.2 Case Name: Security Testing for PAAS

Requirement Description: The system should be able to detect any flaws in confidentiality integrity, authentication, authorisation and non-repudiation.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	(Confidentiality) Enter the URL manually to access to the Create User page	`http://localhost:53861/PGOs/create`	User shouldnt be able to view the content of the Create User (hence dont have access to customer details)	Create User page is displayed.	Fail	Create User page is displayed by bypassing the login in screen so page is visible to the user without login in.
2	(Integrity) Enter the URL manually to access to the user edit page	`http://localhost:53861/PGOs/Edit/1`	User shouldnt be able to edit the content of a user without logging in	User Edit page is displayed	Fail	User Edit page has been shown to the user, so the user can edit the user details without login in
3	(Authentication) Enter the correct login details on the Login page and click on Login button	Email Address: `admin@admin.co.za` Password: admin	Login details are accepted and user can view the Staff Portals page	The system does as expected.	Pass	
4	(Authentication) Enter the wrong login details on the Login page and click on Login button	Email Address: `trkye@yahoo.com` Password: pass123	Login details are rejected and user cant view the Staff Portals page	The system does as expected.	Pass	
5	(Authorisation) Enter the URL manually to access to the Application Interviews	`http://localhost:53861/Interviews/ApplicationsInterviews`	User shouldnt be able to see the content of the Application Interviews list without logging in	Application Interviews page is not displayed to the user. User is asked to login.	Pass	
6	(Availability) View the user details in the PGO view and compare it to the PGO details in the database	`http://localhost:53861/PGOs/index`	PGO details in the PGO view page and PGO details in the PGO Table should match	The system does as expected.	Pass	

7	(Non-repudiation) Download the uploaded PDF document in the database.		User should be able to view if the uploaded PDF document has been created in the database	The system does as expected.	Pass	

5.3 User Acceptance Testing (UAT)

5.3.1 Motivation for User Acceptance Testing

The user of the software product performs User Acceptance Testing (UAT). The acceptance criteria is specified for a given scenario to test that certain scenario. This test is important so we can evaluate the software under actual business or real-world scenarios to check if the software meets the requirements requested by the client. If a certain scenario fails the test then the results will be analysed to classify the priority of each scenario.

This testing method will highlight if any feature of the system is not working. The team will use the results to reconfigure these processes.

5.3.2 Case Name: Create Application

Requirement Description: The system should be able to validate the user input and evaluate the given use case.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	Fill-in all the input boxes with valid information in Create Application	Student Number:789654 First Name: James Last Name: Doe ID Number: 9403056132181 Email address: james-doe@gmail.com Contact Number: 0725896321 School: Information Systems Faculty: Science Street Number: 21 Street Name: Inner Street Suburb: Observatory City: Johannesburg Province: Gauteng	The user should be directed to the confirmation page of Create Application.	The system did as expected.	Pass	
2	Do not fill-in any of the input boxes on the Create Application page	Student Number: First Name: Last Name: ID Number: Email address: Contact Number: School: Faculty: Street Number: Street Name: Suburb: City: Province:	The user will be asked to enter the required details with a warning under the empty fields.	The system did as expected.	Pass	

5.3.3 Case Name: Create Interview

Requirement Description: The system should be able to validate the user input and evaluate the given use case.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	Fill-in all the input boxes with valid information in Create Interview	Date: 2017/09/30 Interview Time: 01:30PM Venue: CLM	The user should be directed to the confirmation page of Create Interview.	The system did as expected.	Pass	
2	Do not fill-in any of the input boxes on the Create Interview Page	Date: Interview Time: Venue:	The user will be asked to enter the required details with a error warning under the empty fields.	The system did as expected.	Pass	
2	Do not fill-in any of the input boxes on the Create Application page	Student Number: First Name: Last Name: ID Number: Email address: Contact Number: School: Faculty: Street Number: Street Name: Suburb: City: Province:	The user will be asked to enter the required details with a warning under the empty fields.	The system did as expected.	Pass	

5.4 Cross-Browser Compatibility Testing

5.4.1 Motivation for Cross-Browser Compatibility Testing

Cross browser testing is the process of testing a web application across different browsers to ensure that a web application works as intended across multiple browsers since certain components might work differently on different web browsers.

We shall test our web application on the following web browsers:

- Microsoft Edge version 40.15063.0.0
- Google Chrome version 60.0.3112.113
- Mozilla Firefox version 55.0.3

Desktop Browser tests have been conducted on Windows 10 operating system.

Mobile Browser tests have been conducted on Android 7.0 Nougat operating system.

5.4.2 Case Name: Check Cross-Browser Compatibility

Requirement Description: The system should be compatible with different types of browsers.

Test Number	Action/ Task	Test Input	Expected Results	Actual Results	Pass/Fail	Comments
1	Access the Web Application using the Google Chrome Desktop browser (version 60.0.3112. 113)	Run the PAAS Web Application on Google Chrome Desktop Browser (version 60.0.3112. 113)	The user should be able to view and interact with the PAAS website, including login into the system	The system did as expected	Pass	
2	Access the Web Application using the Google Chrome Mobile browser (version 60.0.3112. 116)	Run the PAAS Web Application on Google Chrome Mobile Browser (version 60.0.3112. 116)	The user should be able to view and interact with the PAAS website, including login into the system	The system did as expected	Pass	
3	Access the Web Application using the Mozilla Firefox Desktop browser (version 55.0.3)	Run the PAAS Web Application on Mozilla Firefox Desktop Browser (version 55.0.3)	The user should be able to view and interact with the PAAS website, including login into the system	The system did as expected	Pass	
4	Access the Web Application using the Mozilla Firefox Mobile browser (version 55.0.2)	Run the PAAS Web Application on Mozilla Firefox Mobile Browser (version 55.0.2)	The user should be able to view and interact with the PAAS website, including login into the system	The system did as expected	Pass	
5	Access the Web Application using the Microsoft Edge browser (version 40.15063.0.0)	Run the PAAS Web Application on Microsoft Edge browser (version 40.15063.0.0)	The user should be able to view and interact with PAAS website, including login into the system	Failed to display date/time picker	Fail	

5.5 Test Summary

Functionality Testis was successful for all specified use cases. The system was able to achieve all the requests from the users side. The system was able to create an application, download a document, create an interview, recommend an application and login users.

Security Test was successful in authenticating the user when the user enters the correct and incorrect details. System can detect if the user should be allowed to access or not. Users may not manually access via the URL. Logged in users may be able to download the files requested.

User Acceptance Testing has been conducted on specified use cases. The validity of the input details are checked. The user may not leave any boxes unfilled.

Cross Browser Compatibility Test was successful for all the specified browsers except Microsoft Edge. Microsoft Edge does not display the Date Picker. So, the user cannot view Date Picker hence they cant select a date. This feature works well with other browsers. After analyses and research, we have found that Microsoft Edge has default style and selector for date. The default style takes precedence over additional styling done through the plugin. The user can still enter a date manually with the format of yyyy/mm/dd. This is problematic as the does not know the format hence will not be able to enter the correct date format. This issue is noted and will be fixed in construction phase.

References

[1] Clark Wimberly, sitepoint,
https://www.sitepoint.com/5-simple-ux-principles-guide-product-design/

[2] Scrum Alliance,
https://www.scrumalliance.org/why-scrum

[3] Bob Pepalis, IZENDA,
https://www.izenda.com/blog/5-benefits-3-tier-architecture/

[4] IBM Knowledge Centre,
https://www.ibm.com/support/knowledgecenter/en/SSAW57_8.5.5/com.ibm.websphere.nd.doc/ae/covr_3-tier.html