



Republic of the Philippines
CAVITE STATE UNIVERSITY
Bacoor City Campus
SHIV, Molino VI, City of Bacoor
□ (046) 476-5029

Enrollment System for Department of Computer Studies – Cavite State University – Bacoor Campus

Bides, Matthew Isaac L.

Bustero, John Michael C.

Dasalla, Keith Gabriell F.

Fernandez, Alexander James Ian J.

Galvez, Dioren Noah M.

Labalan, Jerald V.

Muega, Michael Angelo I.

Orcullo, James Andrei V.

January 2025

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

TABLE OF CONTENTS

Summary	3
Introduction	4
Objectives	5
Scope and Limitations	6
Major Contributions or Roles of Team Members	8
System Analysis	10
Requirements Analysis	10
Use Case Analysis	14
Process Flow	22
System Design	23
Architecture	23
Database Design	25
User Interface Design	31
Implementation	39
Development Tools and Environment	39
Features and Functionalities	40
Testing	48
Test Plan	48
Test Cases	48
Unit Testing	49
Maintenance	64
Conclusion	67
Appendices	68
Glossary of Terms	68

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

References	64
Additional Diagrams and Tables	70
Agile SDLC	70
Gantt Chart	71

SUMMARY

The Cavite State University - Bacoor City Campus Department of Computer Studies Enrollment System is a web application developed to address the primary challenges identified during requirements gathering for Information Technology and Computer Science students residing on campus.

The enrollment system serves as a strategic solution to the challenges faced in previous semesters, such as managing large volumes of students and ensuring the timely dissemination of information. The system will benefit the following:

1. **CvSU-Bacoor Information Technology and Computer Science students.** The students can benefit from a faster, more efficient enrollment process that minimizes delays and overcrowding during enrollment periods, as each student is assigned a specific schedule for advising and enrollment.
2. **CvSU-Bacoor Department of Computer Studies (DCS).** The department will benefit from having students' records digitized, allowing them to be easily managed and exported as spreadsheets. Additionally, the system will help minimize crowding during the advising process and counterchecking of requirements.
3. **Student organizations,** such as the **Alliance of Computer Scientists (ACS)** and the **Information Technology Society (ITS),** can benefit from the ability to track and update enrollees' statuses during the enrollment process. In addition, the system can issue receipts for society fees once payments are made. Records regarding the number of enrolled students can also be updated directly, eliminating the need to manually create data forms each semester, as information can be exported directly from the system.
4. **Registrar staff.** The system simplifies the enrollment process, including the issuance of pre-enrollment forms and certificates of registration. The subjects listed on the certificate of

registration are those selected by the students, under the supervision of their adviser, during the advising period.

5. CvSU-Bacoor administration heads. The administration can monitor enrollment stats, and streamline their administrative tasks.

INTRODUCTION

The management of paperwork for submitting requirements in higher education is increasingly adapting to a more technology-oriented approach. Systems are being developed and implemented to address numerous problems encountered during enrollment. The overcrowding of students, increased workloads, and counterproductive tasks have become significant challenges. The Cavite State University - Bacoor City Campus' Department of Computer Studies is currently facing these issues. As the focal point for the university's technological management, the department is lagging behind and encountering additional problems. To address these concerns, the project aims to create a more effective system that streamlines the enrollment process and document management. The web application serves as a central hub for collecting, processing, and accessing student information.

The system offers several features (see page 10, Table 2) that enhance the enrollment process and set it apart from the traditional enrollment setting. Through the system, students can **view the enrollment status, check enrollment dates, submit curriculum checklist** (for counterchecking), **select advising dates and subjects** based on the designed academic workload, and **download a copy of Certificate of Registration** for purposes such as scholarship or part-time job applications.

Administrators (society executives and registrar staff) can **create enrollment periods, add and edit user roles** (such as advisers and society officers), **export enrollee/student data, and print Certificate of Registration.**

Society officers can **update the enrollee's status** based on the stations completed during the face-to-face enrollment process, as well as **issuance of receipt** once the society fee has been paid.

Advisers should be able to **accept or reject scanned copies of documents**, such as **Signed Curriculum Checklists, Transcripts of Records**, and the previous semester's **Certificate of Registration**, regardless of whether the original physical copies are submitted in person. Once verified that the student was a bona fide student in the previous semester or is seeking admission to the university, advisers can review the enrollee's desired subjects and modify them as necessary through negotiation or advising during the scheduled advising session. Once approved, the student will be assigned an official enrollment date by the designated admin (either society presidents or secretariat).

OBJECTIVES

The objective of the project is to develop a comprehensive enrollment system for the Department of Computer Studies (DCS) at Cavite State University - Bacoor City Campus. This system aimed to streamline and enhance the face-to-face enrollment process, making it more efficient and accessible for both students and administrative staff. Additionally, the project aimed to achieve the following specific objectives:

- Identify the current issues faced by students, society officers, and registrar staff by conducting face-to-face interviews (see Appendix).
- Analyze the challenges and limitations of the existing face-to-face enrollment process and explore the potential benefits of implementing a new system.
- Design the system using the Agile Scrum methodology.
- Develop a web application using Visual Studio Code 2024 as the IDE, incorporating system functions that utilize React.js and Node.js/Express.js for

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

creating front-end and back-end functionalities, and MySQL as the database; use Canva Pro 2024 and Figma for application design; Google Docs for documentation, and draw.io, Creately, and Lucidchart for creating diagrams.

SCOPE AND LIMITATIONS

The enrollment system for the Department of Computer Studies (DCS) at Cavite State University - Bacoor City Campus is designed to enhance the enrollment process by incorporating features such as **enrollment status tracking, document submission, advising date and enrollment date selection, Certificate of Registration, pre-enrollment form and payment receipt issuance and generation**. This system aims to provide students and administrative staff with an efficient and user-friendly platform for managing enrollment activities. A web application has been developed for students to access their enrollment information and submit required documents, while a separate interface is available for society officers, advisers and registrar staff to manage enrollment data and user accounts.

The system functionality includes **user role management, real-time updates on enrollment status, document verification by advisers, and reporting features for all the enrollment team** to analyze enrollment trends. Students can electronically **submit curriculum checklists, Transcript of Records, and Last Semester's Certificate of Registration**, allowing them to view important enrollment dates and advising schedules. Additionally, admins (such as society executives and registrar staff), are able to **post announcements** that can be seen via landing pages and notifications.

The implementation of the enrollment system depends on several factors, including **digital literacy, access to devices, and internet connectivity**, which may affect student engagement and usage. Ongoing system maintenance and updates are essential, requiring consistent support and resources to ensure long-term sustainability. Data privacy and security are critical, necessitating measures to protect user information and comply with relevant data protection regulations.

However, the system has certain limitations. It **does not support offline access**, as it is a real-time web application that requires an internet connection for full functionality. Minimum requirements for the system include a modern web browser and an active internet connection. Recommended requirements include a device with updated software and sufficient processing power to ensure smooth operation. Hardware requirements include a stable internet connection, and users must have access to devices with adequate specifications to utilize all functionalities effectively. Features such as document uploads and real-time updates are contingent upon users having the necessary technology and connectivity.

Additionally, processes such as the submission of necessary documents (Transcript of Records, Admission Form, Shifting Form, and Equivalency Form), even though some may be submitted online, and the **selection of subjects are still required to be completed face-to-face**, depending on the enrollees' advising date. Due to protocols established by the Office of Student Affairs (OSAS) and the organizations' coordinators, **online payment for society membership fees is not permitted**, necessitating that this process also be conducted in person.

ROLES OF TEAM MEMBERS

In the development of the enrollment system, each team member had specific tasks and responsibilities that contributed to the project's success. Below are the major contributions and roles of the team members.

Name	Role	Major Contribution
Bides, Matthew Isaac L.	Back-end Developer Database Administrator	Developed the whole server-side logic and functionality of the system. Managed and maintained the organization's database.
Bustero, John Michael C.	Front-End Developer	Created the user interface based on the design. Developed the user interface on the dashboard for all user roles, and user interface for adviser and society officer side.
Dasalla, Keith Gabriell F.	Project Manager UI/UX Designer Documentation Specialist	Schedule the tasks, and oversaw the project timeline and the quality of the project. Conducted and assessed the requirement specifications based on user needs and created the wireframes and prototype of the system. Compiled and organized the project documentation, for future maintenance and operations Created diagrams such as sequence diagrams and use case diagrams.
Fernandez, Alexander James Ian J.	QA Tester Back-end Developer	Conducted testing on the front-end side of the system to identify bugs. Also helped in passing all the test cases needed for documentation.

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

		<p>Helped in developing the server-side logic of the system, and the connection to its database.</p> <p>Created the Data Flow Diagram of the system</p>
Galvez, Dioren Noah M.	QA Tester ▾ Front-End Developer ▾	<p>Conducted testing on the back-end side of the system.</p> <p>Helped in creating the front-end functionality of the announcements and notifications.</p>
Labalan, Jerald V.	Asst. Project Manager ▾ Front-End Developer ▾	<p>Supports the project manager in decision making, as well as answering queries in terms of the creation of documentation, and development.</p> <p>Developed and refined the user interface to create a more uniform look and presentation, while also adding new features to the system.</p> <p>Created the Entity Relationship Diagram of the System.</p>
Muega, Michael Angelo I.	Front-End Developer ▾	Developed the adviser, and enrollee user interface.
Orcullo, James Andrei V.	Front-End Developer ▾	Developed the adviser, and enrollee user interface.

SYSTEM ANALYSIS

REQUIREMENT ANALYSIS

FUNCTIONAL REQUIREMENTS

- **User Registration.** Allows the user to register an account, by providing necessary information.
- **Announcements and Notifications.** Allows administrators to post important announcements, which can be visible through the landing page, notifications and enrollee dashboard.
- **Enrollment Team Dashboard.** Interface for the enrollment team to manage enrollment trends and updates. It includes the number of students enrolled each program, number of students paid on society fee membership, and number of regular students (S1 and S2) and irregular students (S3, S4 and S5)
- **Enrollee Management/Enrollee Team Controls.** Commands for managing the details and status of enrollees, which are controlled by the enrollee team (system administrator, society officers and advisers).
- **Enrollment Team Management/Admin Only Controls.** Management of the members of the enrollment team.
- **Print Certificate of Registration.** Allows users to print the Certificate of Registration once enrolled in the university.
- **Print Society Fee Receipt.** Allows society officers to print the receipt once the enrollee is paid.
- **Print Pre-Enrollment Form.** Allows users to print the Pre-Enrollment Form after the scheduled advising date.
- **Advising and Enrollment Date Management.** Enrollees can select an advising date, while the Society Officers will select the enrollment date after the enrollees advising

session.

- **Enrollment Queuing Status Management.** Society officers can update the enrollee's status during the face-to-face enrollment.
- **Requirement Submissions.** Enrollees are able to submit requirements via the file upload option.
- **Enrollee Profile Management.** Manages profile for enrollees.
- **Export Records.** allows the enrollment team to export records for reporting and analysis.
- **Enrollment Guide.** Includes guides for enrollment and Frequently Asked Questions (FAQs).

The table below are the user classes and its accessible features in the system.

User Classes	User Role	Accessible Features
System Administrator (Registrar Staff and Society President and Secretariat)	Responsible for overseeing, managing of student records	<ol style="list-style-type: none"> 1. Login Account <ul style="list-style-type: none"> 1.1 Verify Login Credentials 1.2 Display Error Message 2. Set Enrollment Period <ul style="list-style-type: none"> 2.1 Set Regular and Incoming First Year Student Enrollment Period 2.2 Set Irregular (Irregulars, Shiftees, Returnees, Transferees) Student Enrollment Period 2.3 Set Semester and Academic Year 3. Account Management <ul style="list-style-type: none"> 3.1 View Student and Enrollee List <ul style="list-style-type: none"> 3.1.1 Change ‘Enrollee’ to ‘Student’ Status <ul style="list-style-type: none"> 3.1.1.1 Mark as Enrolled 3.2. Manage Adviser and Society Officers Accounts <ul style="list-style-type: none"> 3.2.1 Add Account 3.2.2 Edit Account 3.2.2.3 Edit Role 3.2.3 Delete Account 3.3 Manage Enrollee Account

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

		<p>3.2.3 Delete Account</p> <p>4. Enrollment Data Management</p> <ul style="list-style-type: none"> 4.1 View Responses <ul style="list-style-type: none"> 4.1.1 Export .xlsx Copy of Responses 4.2 View File Upload <p>5. Print ‘Certificate of Registration’</p>
Society Officers	Responsible for managing students' current enrollment status in queue	<p>1. Login Account</p> <ul style="list-style-type: none"> 1.1 Verify Login Credentials 1.2 Display Error Message <p>2. Enrollment Status Management</p> <ul style="list-style-type: none"> 2.1 View Data Form Responses 2.2 Set Society Fee Status 2.3 Set Counterchecking and Advising Status 2.4 Set Issuance of Certificate of Registration Status 2.5 Set Enrollee to Student Status <p>3. Enrollment Date Queueing Management</p> <ul style="list-style-type: none"> 3.1 Schedule Enrollee Enrollment Date <p>4. Print Receipt</p>
Adviser	Responsible for counterchecking signed checklists, TOR, certificate of grades, and subjects selected by students before advising.	<p>1. Login Account</p> <ul style="list-style-type: none"> 1.1 Verify Login Credentials 1.2 Display Error Message <p>2. Requirement Submission Management</p> <ul style="list-style-type: none"> 2.1 View ‘Curriculum Checklist’ 2.2 View ‘Last Semester Certificate of Registration’ 2.3 Reject File Submissions <ul style="list-style-type: none"> 2.3.1 Reject File Submission for ‘Curriculum Checklist’ 2.3.2 Reject File Submission for ‘Last Semester Certificate of Registration’ 2.3.3 Reject File Submission for Both 2.3.4 Add Reason/Feedback for Rejection 2.4 Accept File Submissions <p>*Once accepted, the Adviser can check the tentative and proposed selected subjects by the students*</p> <p>3. Advisee Subject Selection Management</p> <ul style="list-style-type: none"> 3.1 Edit Subjects 3.2 View Pre Enrollment Form

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Enrollees	The primary user of the system, responsible for submitting requirements and applications as well as schedule advising dates.	<ul style="list-style-type: none"> 1. Login Account <ul style="list-style-type: none"> 2.1 Verify Login Credentials 2.2 Display Error Message 2. Account Registration <ul style="list-style-type: none"> 2.1 Fill Up Registration Form <ul style="list-style-type: none"> 2.1.1 Enter Email Address 2.1.2 Enter Password 2.2 Bypass Email Account Verification <ul style="list-style-type: none"> 2.2.1 Enter OTP 3. Profile/Data Form Management <ul style="list-style-type: none"> 3.1 Fill Out Credentials <ul style="list-style-type: none"> 3.1.1 Fill Out ‘Personal Information’ 3.1.2 Fill Out ‘Family Background’ 3.1.3 Fill Out ‘Educational Attainment’ 4. Student Categorization <ul style="list-style-type: none"> 4.1 View Frequently Asked Questions 4.2 Fill Out ‘Student Type’ <ul style="list-style-type: none"> 4.2.1 Select ‘Student Categorization’ 4.2.2 Select ‘Program’ 5. Requirement Submission <ul style="list-style-type: none"> 5.1 Upload ‘Scanned Transcript of Records’ 5.2 Upload ‘Scanned Signed Curriculum Checklist’ 5.3 Upload ‘Last Semester Certificate of Registration’ 6. Tentative Subject Selection <ul style="list-style-type: none"> 6.1 Select Subjects 7. Advising Schedule Management <ul style="list-style-type: none"> 7.1 Select Advising Date 8. Enrollment Date Queueing Management <ul style="list-style-type: none"> 6.1 Appeal Desired Enrollment Date 9. Enrollment Tracking <ul style="list-style-type: none"> 9.1 View Enrollment Status 10. Print ‘Certificate of Registration’

NON-FUNCTIONAL REQUIREMENTS

The following are the non-functional requirements of the new enrollment system:

3.1 Reliability

3.1.1 The system should be reliable, with minimal disruptions.

3.2 Security

3.2.1 Only authorized users (enrollees, admins and society officers) can access the system, as it utilizes approval of data form entries for enrollees and user credentials will be sent personally for system administrators.

3.2.2 The system should adhere to data privacy and security policies, ensuring that data is encrypted to maintain its confidentiality and integrity.

3.3 Performance

3.3.1 The system must handle the multiple user operations at once, during the enrollment process.

3.4 Usability

3.4.1 The system should be easy to navigate, and serves its purpose within its intended audience.

3.4.2 The system should follow the desired flow of the enrollment process based on the responses collected from the interviews.

USE CASE ANALYSIS

The use case depends on which student type. In the system, there are six student types, and have different interactions within the system.

Use Case Specification	Name of Use Case	Regular Student (S1)	Written by	Dasalla, Keith Gabriell	Date	11/11/24	
			Updated by	Dasalla, Keith Gabriell	Date	1/5/25	
Overview	<p>This use case diagram describes the interaction of regular enrollees (S1) with the enrollment team. The student has the ability to register an account and fill out all the information required by the system administrators, advisers, and society officers. A student will only be recognized as a regular student if they take the normal academic load, and passed all the courses last semester.</p>						
Eventflow	<p>Main Flow</p> <ol style="list-style-type: none"> 1. The student registers an account by adding their email and password. Once the student has filled out the necessary information (IDs 3-5), the admins, advisers, and society officers may view the data. <p>Alternative Flow</p> <ol style="list-style-type: none"> 1. If the student already has an account and has filled out the necessary information during registration (IDs 3-5), the student is allowed to log in to their account. 2. The society officer will then schedule the enrollment date, which can be viewed by the student. <p>Remarks</p> <ol style="list-style-type: none"> 1. Regular students will submit a scanned copy of Last Semester's Certificate of Registration and Curriculum Checklist signed by the advisers. Aside from that, it will be submitted on the scheduled enrollment date. 						

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Relationship	<pre> graph TD Student((Student)) -- "<<extend>>" --> AccountRegistration[2. Account Registration] Student -- "<<extend>>" --> LoginAccount[1. Login Account] Admin((Admin)) --> ViewEnrollmentDataResponses[4. View Enrollment Data Responses] Adviser((Adviser)) --> ViewRequirementSubmissions[2. View Requirement Submissions] SocietyOfficer((Society Officer)) -- "<<extend>>" --> ViewEnrollmentDate[View Enrollment Date] ViewEnrollmentDataResponses --> ViewRequirementSubmissions ViewRequirementSubmissions --> ViewEnrollmentDate ViewEnrollmentDate -- "<<include>>" --> ViewEnrollmentDataResponses ViewEnrollmentDataResponses --> ViewEnrollmentDataResponses ViewEnrollmentDataResponses --> ViewRequirementSubmissions ViewEnrollmentDataResponses --> ViewEnrollmentDate ViewRequirementSubmissions --> ViewEnrollmentDataResponses ViewRequirementSubmissions --> ViewEnrollmentDate ViewEnrollmentDate --> ViewEnrollmentDataResponses ViewEnrollmentDate --> ViewRequirementSubmissions ViewEnrollmentDate -- "<<extend>>" --> ScheduleEnrollmentDate[3.1 Schedule Enrollee Enrollment Date] ScheduleEnrollmentDate --> ViewEnrollmentDataResponses ScheduleEnrollmentDate --> ViewRequirementSubmissions ScheduleEnrollmentDate --> ViewEnrollmentDate </pre>
Precondition	<ol style="list-style-type: none"> 1. All user roles have access to a device with an Internet connection. 2. All user roles have logged in into the system. 3. The student must fill in the necessary information (IDs 3-5) before having a scheduled enrollment date.

Use Case Specification	Name of Use Case	Irregular Student (S3)	Written by	Dasalla, Keith Gabriell	Date	11/12/24
			Updated by	Dasalla, Keith Gabriell	Date	1/5/25
Overview	<p>This use case diagram describes the interaction of irregular enrollees (S3) with the enrollment team. The student has the ability to register an account and fill out all the information required by the system administrators, advisers, and society officers. A student will only be recognized as an irregular student if they resided on campus during the last semester and did not take the normal academic load.</p>					
Eventflow	<p>Main Flow</p> <ol style="list-style-type: none"> 1. The student registers an account by adding their email and password. Once the student has filled out the necessary information (IDs 3-5), the admins, advisers, and society officers may view the data. <p>Alternative Flow</p> <ol style="list-style-type: none"> 1. If the student already has an account and has filled out the necessary information during registration (IDs 3-5), the student is allowed to log in to their account. 2. The student must select an advising date along with their desired subjects for the semester. 3. The adviser can view the student's submission of the proposed subjects. 4. After the advising session, the adviser will change the proposed subjects to better align with the student's academic load. 					

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

	<p>5. The society officer will then schedule the enrollment date, which can be viewed by the student.</p> <p>Remarks</p> <ol style="list-style-type: none"> Irregular students will submit a scanned copy of Last Semester's Certificate of Registration and Curriculum Checklist signed by the advisers. Aside from that, it will be submitted on the scheduled enrollment date.
Relationship	<pre> usecaseDiagram actor Student actor Admin actor Adviser actor SO [Society Officer] usecase UC1 [1. Login Account] usecase UC2 [2. Account Registration] usecase UC3 [3. Fill Out Profile Data Form] usecase UC4 [4. Fill Out Student Categorization] usecase UC5 [5. Submit Last Semester COR and Checklist] usecase UC6 [Select Advising Date and Subjects] usecase UC7 [View Enrollment Date] usecase UC8 [2. View Requirement Submissions] usecase UC9 [2.1 View Data Form Responses] usecase UC10 [3.1 Schedule Enrollee Enrollment Date] Student --> UC1 : <<extend>> Student --> UC2 : <<extend>> Student --> UC3 : <<extend>> Student --> UC4 : <<extend>> Student --> UC5 : <<extend>> Student --> UC6 : <<extend>> Admin --> UC4 : <<extend>> Admin --> UC5 : <<extend>> Admin --> UC8 : <<extend>> Admin --> UC9 : <<extend>> Admin --> UC10 : <<extend>> Adviser --> UC8 : <<extend>> Adviser --> UC9 : <<extend>> SO --> UC10 : <<extend>> UC1 --> UC2 : <<extend>> UC2 --> UC3 : <<extend>> UC3 --> UC4 : <<extend>> UC4 --> UC5 : <<extend>> UC5 --> UC6 : <<extend>> UC6 --> UC7 : <<extend>> UC7 --> UC8 : <<extend>> UC8 --> UC9 : <<extend>> UC9 --> UC10 : <<extend>> UC6 --> UC8 : <<include>> UC6 --> UC9 : <<include>> </pre>
Precondition	<ol style="list-style-type: none"> All user roles have access to a device with an Internet connection. All user roles have logged in into the system. The student must fill in the necessary information (IDs 3-5) before selecting an advising date and subjects The student must be advised to get the enrollment date.

Use Case Specification	Name of Use Case	Incoming First Year (S2)	Written by	Dasalla, Keith Gabriell	Date	11/13/24
		Updated by	Dasalla, Keith Gabriell	Date	1/5/25	
Overview	This use case diagram describes the interaction of incoming first year enrollees (S2) with the enrollment team. The student has the ability to register an account and fill out all the information required by the system administrators, advisers, and society officers.					
Eventflow	<p>Main Flow</p> <ol style="list-style-type: none"> The student registers an account by adding their email and password. Once the student has filled out the necessary information (IDs 3-5), the admins, advisers, and society officers may view the data. 					

	<p>Alternative Flow</p> <ol style="list-style-type: none"> 1. If the student already has an account and has filled out the necessary information during registration (IDs 3-4), the student is allowed to log in to their account. 2. The society officer will then schedule the enrollment date, which can be viewed by the student. <p>Remarks</p> <ol style="list-style-type: none"> 1. There are no submissions of requirements for incoming first year students (S2), as all the requirements were submitted prior to the enrollment period via portal, and the Registrar's Office.
Relationship	<pre> graph TD subgraph IFYS [Incoming First Year Students] S1[1. Login Account] --> S2[2. Account Registration] S2 --> S3[3. Fill Out Profile Data Form] S3 --> S4[4. Fill Out Student Categorization] S4 --> S5[View Enrollment Date] S5 --> S6[4. View Enrollment Data Responses] S6 --> A1[2. View Requirement Submissions] A1 --> A2[2.1 View Data Form Responses] SO1[3.1 Schedule Enrollee Enrollment Date] --> A2 SO1 --> SO2[3.2 View Enrollment Date] Admin --> S6 Adviser --> A1 SO --> SO1 end </pre> <p>The diagram illustrates the interaction flow for incoming first-year students. It starts with a student logging in (1. Login Account). This leads to account registration (2. Account Registration), filling out profile data (3. Fill Out Profile Data Form), and filling out student categorization (4. Fill Out Student Categorization). After these steps, the student views their enrollment date (View Enrollment Date). Finally, the student views their enrollment data responses (4. View Enrollment Data Responses). Additionally, there are interactions with an Admin (viewing enrollment data responses), an Adviser (viewing requirement submissions and data form responses), and a Society Officer (viewing enrollment date and scheduling enrollee enrollment date).</p>
Precondition	<ol style="list-style-type: none"> 1. All user roles have access to a device with an Internet connection. 2. All user roles have logged in into the system. 3. The student must fill in the necessary information (IDs 3-4) before selecting an advising date and subjects

Use Case Specification	Name of Use Case	Transferee Student (S6)	Written by	Dasalla, Keith Gabriell	Date	11/14/24
			Updated by	Dasalla, Keith Gabriell	Date	1/5/25
Overview	<p>This use case diagram describes the interaction of transferee enrollees (S6) with the enrollment team. The student has the ability to register an account and fill out all the information required by the system administrators, advisers, and society officers. A student will only be recognized as a transferee if they resided on other CvSU campuses as a regular student transferring from the same program.</p>					
Eventflow	<p>Main Flow</p>					

	<p>1. The student registers an account by adding their email and password. Once the student has filled out the necessary information (IDs 3-5), the admins, advisers, and society officers may view the data.</p> <p>Alternative Flow</p> <ol style="list-style-type: none"> If the student already has an account and has filled out the necessary information during registration (IDs 3-4), the student is allowed to log in to their account. The society officer will then schedule the enrollment date, which can be viewed by the student. <p>Remarks</p> <ol style="list-style-type: none"> Transferees will submit a scanned copy of Transcript of Records. Aside from that, it will be submitted on the scheduled advising date.
Relationship	<pre> graph TD Student((Student)) -- "<<extend>>" --> AccountRegistration[2. Account Registration] Student -- "<<extend>>" --> LoginAccount[1. Login Account] Admin((Admin)) -- "<<extend>>" --> ViewResponses[4. View Enrollment Data Responses] Adviser((Adviser)) -- "<<extend>>" --> ViewSubmissions[2. View Requirement Submissions] SO((Society Officer)) -- "<<extend>>" --> ViewDataFormResponses[2.1 View Data Form Responses] SO -- "<<extend>>" --> ScheduleEnrollment[3.1 Schedule Enrollee Enrollment Date] ViewResponses --> ViewSubmissions ViewResponses --> ViewDataFormResponses ViewResponses --> ScheduleEnrollment ViewSubmissions --> ViewDataFormResponses ViewSubmissions --> ScheduleEnrollment ViewDataFormResponses --> ScheduleEnrollment </pre> <p>The diagram illustrates the workflow for Transferee Students. It starts with a Student role. The student can either extend the process to '2. Account Registration' or '1. Login Account'. From '1. Login Account', the student can extend to 'Select Advising Date and Subjects' or 'View Enrollment Date'. From 'View Enrollment Date', the student can extend to '3. Fill Out Profile Data Form', '4. Fill Out Student Categorization', or '3.1 Schedule Enrollee Enrollment Date'. Finally, the student can extend to '4. View Enrollment Data Responses' or '2. View Requirement Submissions'. Additionally, an Admin role can extend to '4. View Enrollment Data Responses'. An Adviser role can extend to '2. View Requirement Submissions'. A Society Officer role can extend to both '2.1 View Data Form Responses' and '3.1 Schedule Enrollee Enrollment Date'.</p>
Precondition	<ol style="list-style-type: none"> All user roles have access to a device with an Internet connection. All user roles have logged in into the system. The student must fill in the necessary information (IDs 3-4) before selecting an advising date and subjects The student must be advised to get the enrollment date.
Other information	<ol style="list-style-type: none"> There are no submissions of requirements for transferee students (S6), as all the requirements will be submitted to the Registrars' Office during the advising date.

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Use Case Specification	Name of Use Case	Shiftee and Returnee (S4, S5)	Written by	Dasalla, Keith Gabriell	Date	11/15/24	
			Updated by	Dasalla, Keith Gabriell	Date	1/5/25	
Overview	<p>This use case diagram describes the interaction of shiftee (S4) and returnee (S5) enrollees with the enrollment team. The student has the ability to register an account and fill out all the information required by the system administrators, advisers, and society officers. A student will only be recognized as a shiftee, if the student will shift from another course to IT/CS, or from CS to IT, and vice versa.</p>						
Eventflow	<p>Main Flow</p> <ol style="list-style-type: none"> 1. The student registers an account by adding their email and password. Once the student has filled out the necessary information (IDs 3-4), the admins, advisers, and society officers may view the data. <p>Alternative Flow</p> <ol style="list-style-type: none"> 1. If the student already has an account and has filled out the necessary information during registration (IDs 3-4), the student is allowed to log in to their account. 2. The student must select an advising date along with their desired subjects for the semester. 3. The adviser can view the student's submission of the proposed subjects. 4. After the advising session, the adviser will change the proposed subjects to better align with the student's academic load. 5. The society officer will then schedule the enrollment date, which can be viewed by the student. <p>Remarks</p> <ol style="list-style-type: none"> 1. Transferees will submit a scanned copy of Transcript of Records. Aside from that, it will be submitted on the scheduled advising date. 						

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Relationship	<pre> graph TD Student((Student)) -- "<<extend>>" --> AccountRegistration[2. Account Registration] Student -- "<<extend>>" --> LoginAccount[1. Login Account] Admin((Admin)) --> ViewEnrollmentDataResponses[4. View Enrollment Data Responses] Adviser((Adviser)) --> ViewRequirementSubmissions[2. View Requirement Submissions] SocietyOfficer((Society Officer)) --> ViewDataFormResponses[2.1 View Data Form Responses] SocietyOfficer -- "<<extend>>" --> ScheduleEnrollmentDate[3.1 Schedule Enrollee Enrollment Date] LoginAccount -- "<<extend>>" --> SelectAdvisingDate[Select Advising Date and Subjects] SelectAdvisingDate -- "<<extend>>" --> ViewEnrollmentDate[View Enrollment Date] ViewEnrollmentDate -- "<<extend>>" --> ViewDataFormResponses ViewEnrollmentDataResponses -- "4. View Enrollment Data Responses" --> Admin ViewRequirementSubmissions -- "2. View Requirement Submissions" --> Adviser ViewDataFormResponses -- "2.1 View Data Form Responses" --> SocietyOfficer ScheduleEnrollmentDate -- "3.1 Schedule Enrollee Enrollment Date" --> SocietyOfficer </pre> <p>The diagram illustrates the enrollment process for transferee students. It starts with a student logging in to create an account. This leads to filling out profile and student categorization forms. An administrator views enrollment data responses. An adviser views requirement submissions. A society officer views data form responses and schedules enrollment dates. There are also relationships involving extending the process to include advising date selection and viewing enrollment dates.</p>
Precondition	<ol style="list-style-type: none"> 1. All user roles have access to a device with an Internet connection. 2. All user roles have logged in into the system. 3. The student must fill in the necessary information (IDs 3-4) before selecting an advising date and subjects 4. The student must be advised to get the enrollment date.

PROCESS FLOW

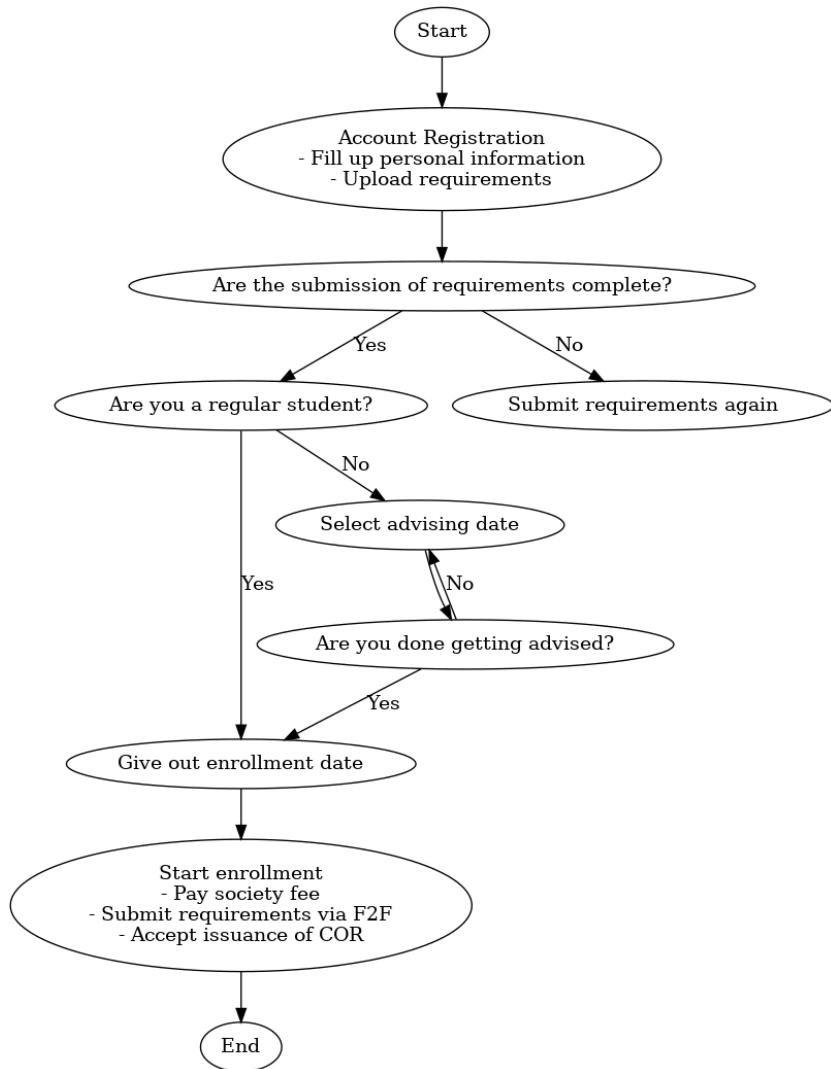


Figure 8. Process Flow of the Enrollment Process

The process flow outlines the model of the system's operations if implemented.

During Account Registration, students must fill out personal information and upload requirements via the system (refer to Enrollees' Accessible Features 2-5, Table 1, page 13). Once completed, irregular students may select an advising date (refer to Enrollees' Accessible Features 7). After advising, the society officer provides the face-to-face enrollment date and initiates the process to change the irregular enrollees' status (refer to Society Officer Accessible Features 2-3, page 12). For regular students, the enrollment

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

process follows a linear path, provided they meet the grade requirements from the previous semester.

SYSTEM DESIGN

ARCHITECTURE

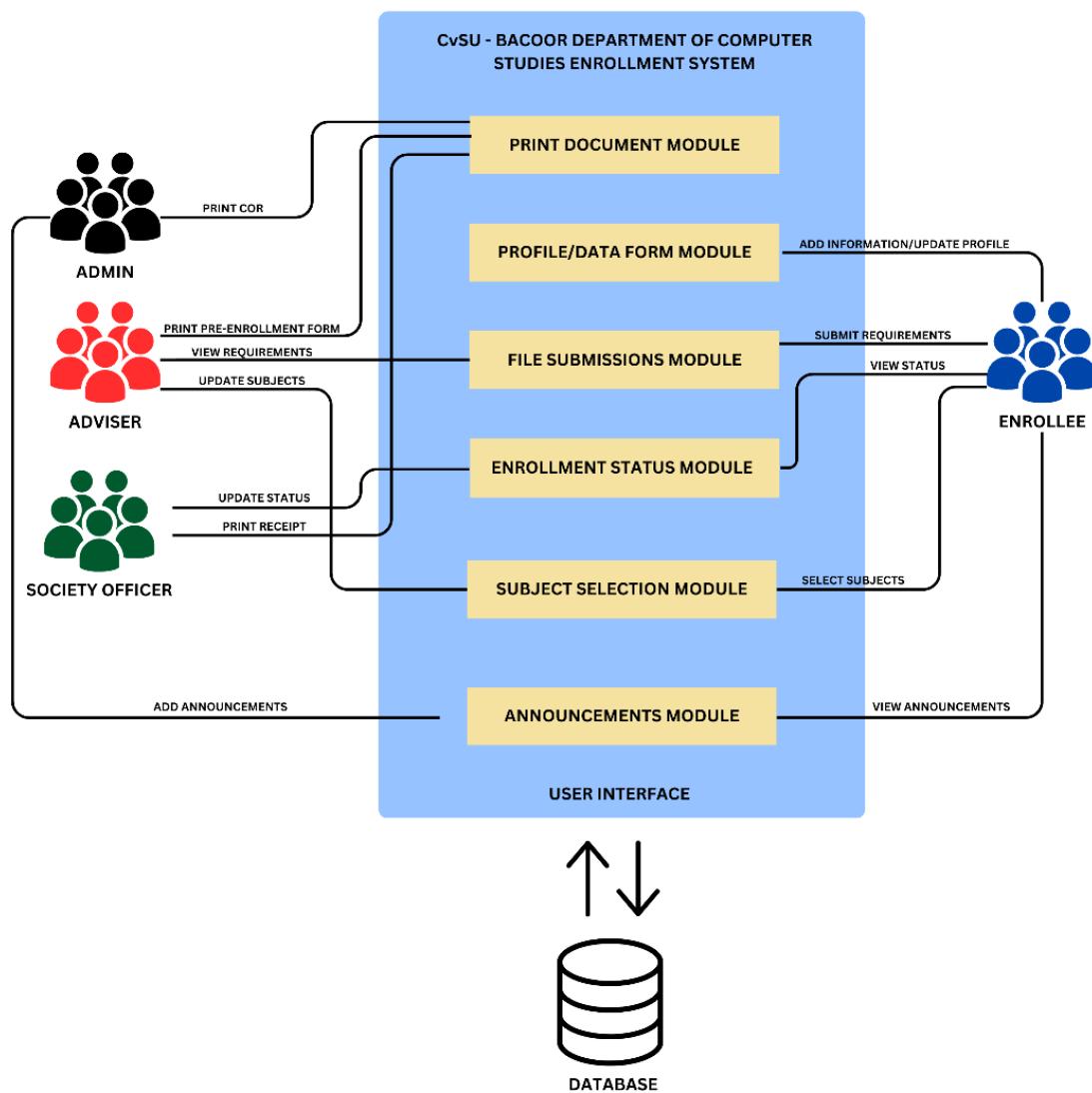


Figure 9. System Architecture of the Project

The enrollment system uses a **layered architecture**. The overall structure of the system consists of the following layers:

User Interface Layer

- The user interface manages user interactions, such as file uploads and action listeners. The system uses React as the framework for creating the front-end interface of the system.

Application Layer

- The application layer is the backend of the system, which processes requests from the User Interface Layer and coordinates with the Database Layer. It is responsible for implementing the system's business logic, handling user requests, user management, and announcement and notification modules. The backend was developed using Node.js with Express.js as the web application framework.

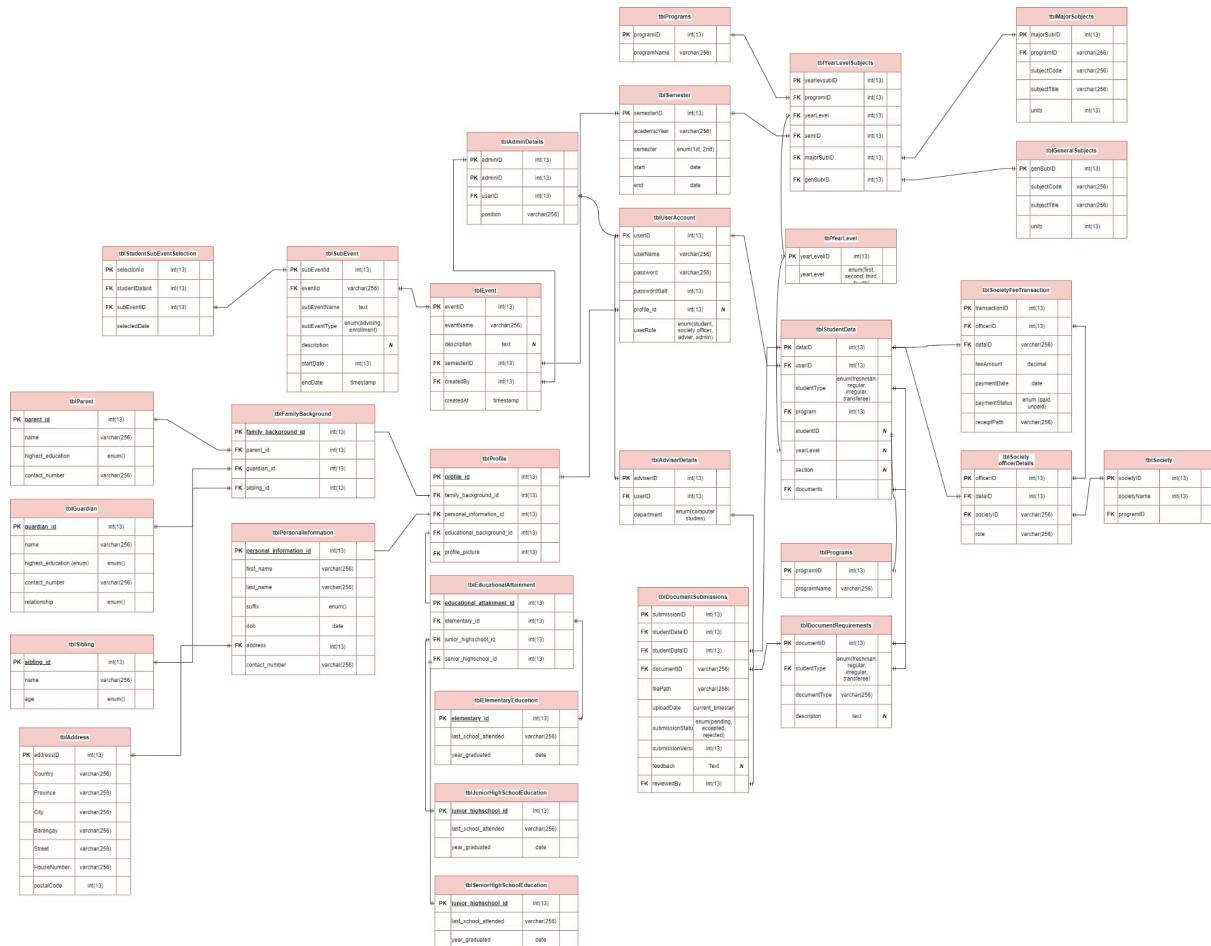
Database Layer

- The database layer stores and retrieves the system's data. The system uses MySQL as the relational database management system for secure and efficient data storage.

Enrollment System for Department of Computer Studies – Cavite State University – Bacoor Campus

DATABASE DESIGN

Entity Relationship Diagram



Entity Relationship Diagram Tables

tblStudentSubEventSelection			
PK	selectionID	int(13)	
FK	studentData	int(13)	
FK	subEventID	int(13)	
	selectedDate	date	

tblSubEvent			
PK	subEventId	int(13)	
FK	eventId	varchar(256)	
	subEventType	enum(advising, enrollment)	
	startDate	date	
	endDate	date	

tblEnrollment			
PK	enrollmentID	int(13)	
FK	semesterID	int(13)	
FK	createdBy	int(13)	
	startDate	date	
	endDate	date	
	createdAt	timestamp	

tblAdminDetails			
PK	adminID	int(13)	
PK	adminID	int(13)	
FK	userID	int(13)	

tblAdviserDetails			
PK	adviserID	int(13)	
FK	userID	int(13)	
	department	enum(computer studies)	

tblYearLevel			
PK	yearLevelID	int(13)	
	yearLevel	enum(first, second, third, fourth)	

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

tblSemester			
PK	semesterID	int(13)	
	academicYear	varchar(256)	
	semester	enum(1st, 2nd)	
	start	date	
	end	date	

tblUserAccount			
PK	userID	int(13)	
	email	varchar(256)	
	password	varchar(256)	
	passwordSalt	int(13)	
	userRole	enum(student, society officer, advier, admin)	

tblYearLevelSubjects			
PK	yearlevsubID	int(13)	
FK	programID	int(13)	
FK	yearLevel	int(13)	
FK	semID	int(13)	
FK	majorSubID	int(13)	
FK	genSubID	int(13)	

tblPrograms			
PK	programID	int(13)	
	programName	varchar(256)	

tblDocumentRequirements			
PK	documentID	int(13)	
FK	studentType	enum(freshman, regular, irregular, transferee)	
	documentType	varchar(256)	
	descripton	text	N

tblAddress			
PK	addressID	int(13)	
	Country	varchar(256)	
	Province	varchar(256)	
	City	varchar(256)	
	Barangay	varchar(256)	
	Street	varchar(256)	
	HouseNumber	varchar(256)	
	postalCode	int(13)	

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

tblGeneralSubjects			
PK	genSubID	int(13)	
	subjectCode	varchar(256)	
	subjectTitle	varchar(256)	
	units	int(13)	

tblMajorSubjects			
PK	majorSubID	int(13)	
FK	programID	varchar(256)	
	subjectCode	varchar(256)	
	subjectTitle	varchar(256)	
	units	int(13)	

tblDocumentSubmissions			
PK	submissionID	int(13)	
FK	studentDataID	int(13)	
FK	documentID	varchar(256)	
FK	reviewedBy	int(13)	
	filePath	varchar(256)	
	uploadDate	current_timestamp	
	submissionStatus	enum(pending, accepted, rejected)	
	submissionVersion	int(13)	
	feedback	Text	N

tblSeniorHighSchoolEducation			
PK	<u>junior_highschool_id</u>	int(13)	
	last_school_attended	varchar(256)	
	year_graduated	date	

tblSocietyOfficerDetails			
PK	userID	int(13)	
FK	officerID	int(13)	

tblProfile			
PK	<u>profile_id</u>	int(13)	
FK	family_background_id	int(13)	
FK	personal_information_id	int(13)	
FK	educational_background_id	int(13)	
	profile_picture		

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

tblSocietyFeeTransaction			
PK	transactionID	int(13)	
FK	officerID	varchar(256)	
FK	studentData	int(13)	
	feeAmount	varchar(256)	
	paymentDate	int(13)	
	paymentStatus	enum()	

tblPersonalInformation			
PK	<u>personal_information_id</u>	int(13)	
FK	address	int(13)	
	first_name	varchar(256)	
	last_name	varchar(256)	
	suffix	enum()	
	dob	date	
	contact_number	varchar(256)	

tblStudentDetails			
FK	userID	int(13)	
FK	program	int(13)	
FK	documents	int(13)	
FK	profileID	int(13)	
	studentID	int(13)	N
	program	enum(cs, it)	
	yearLevel	enum(first, second, third, fourth)	
	studentType	enum(freshman, regular, irregular, transferee)	

tblParent			
PK	<u>parent_id</u>	int(13)	
	name	varchar(256)	
	highest_education	enum()	
	contact_number	varchar(256)	

tblEducationalAttainment			
PK	<u>educational_attainment_id</u>	int(13)	
FK	elementary_id	int(13)	
FK	junior_highschool_id	int(13)	
FK	senior_highschool_id	int(13)	

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

tblJuniorHighSchoolEducation			
PK	<u>junior_highschool_id</u>	int(13)	
	last_school_attended	varchar(256)	
	year_graduated	date	
	year_graduated	date	

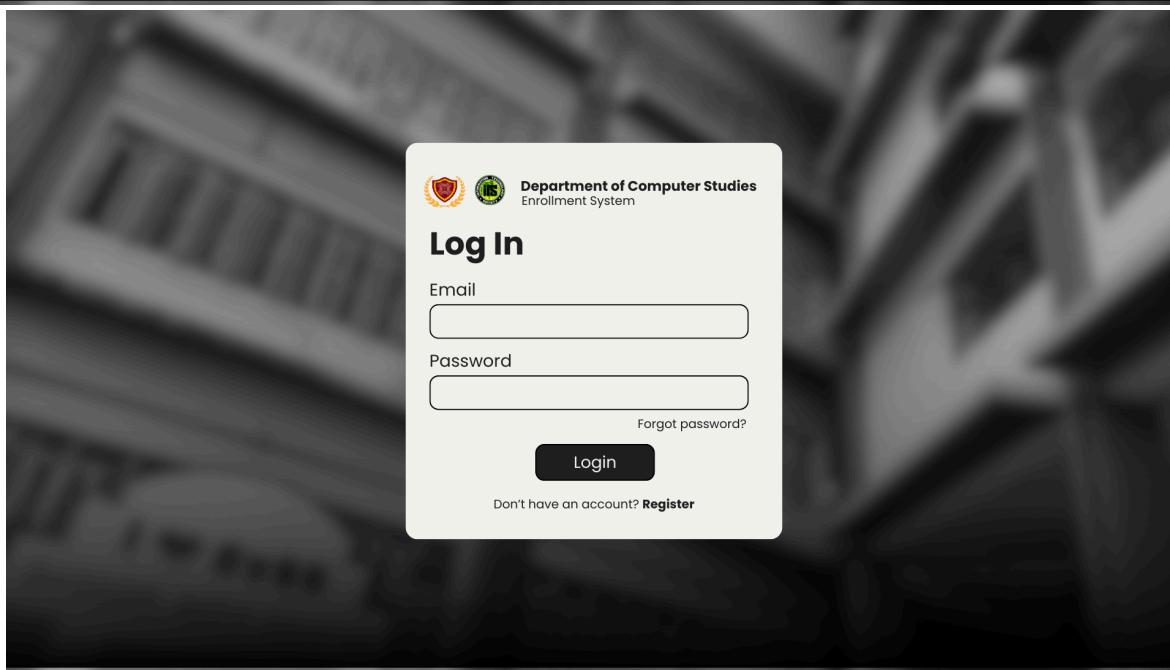
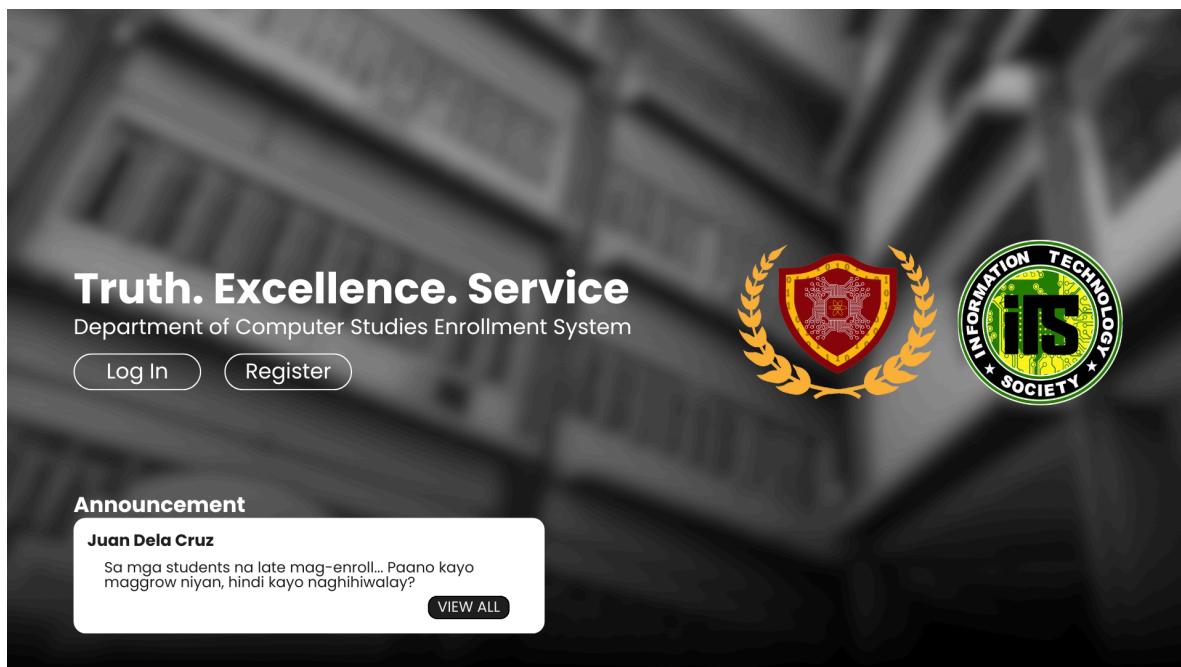
tblSibling			
PK	<u>sibling_id</u>	int(13)	
	name	varchar(256)	
	age	enum()	

tblElementaryEducation			
PK	<u>elementary_id</u>	int(13)	
	last_school_attended	varchar(256)	
	last_school_attended	varchar(256)	
	year_graduated	date	

tblGuardian			
PK	<u>guardian_id</u>	int(13)	
	name	varchar(256)	
	highest_education (enum)	enum()	
	contact_number	varchar(256)	
	relationship	enum()	

USER INTERFACE DESIGN

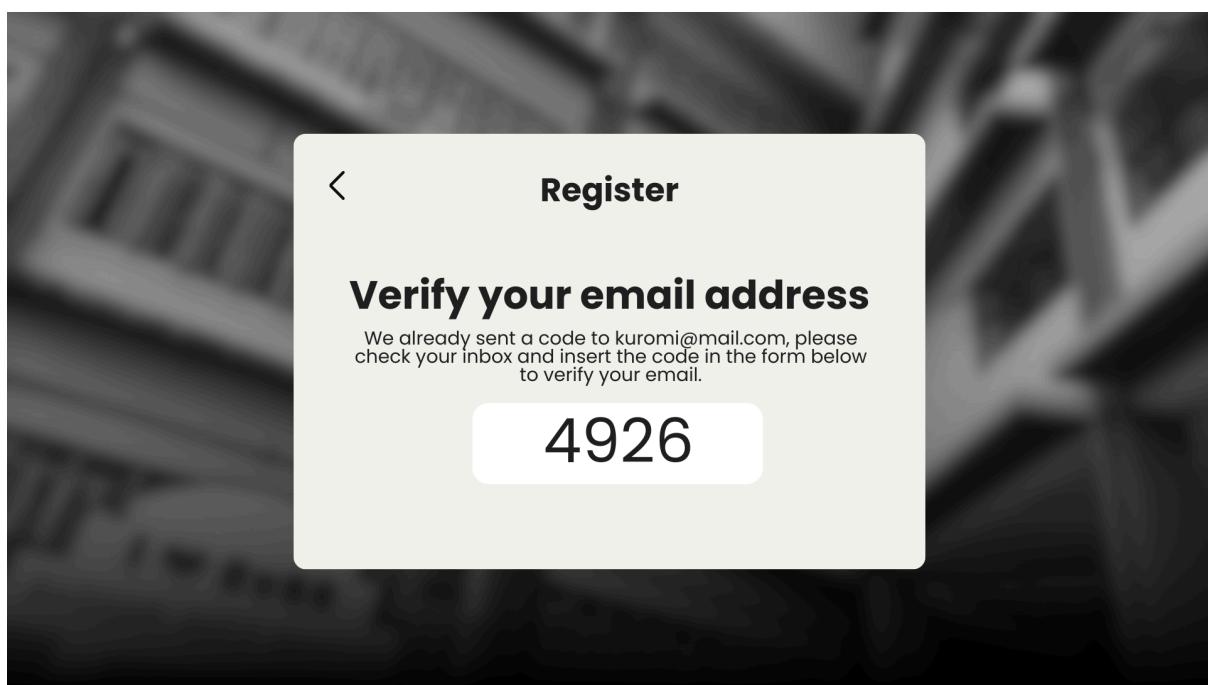
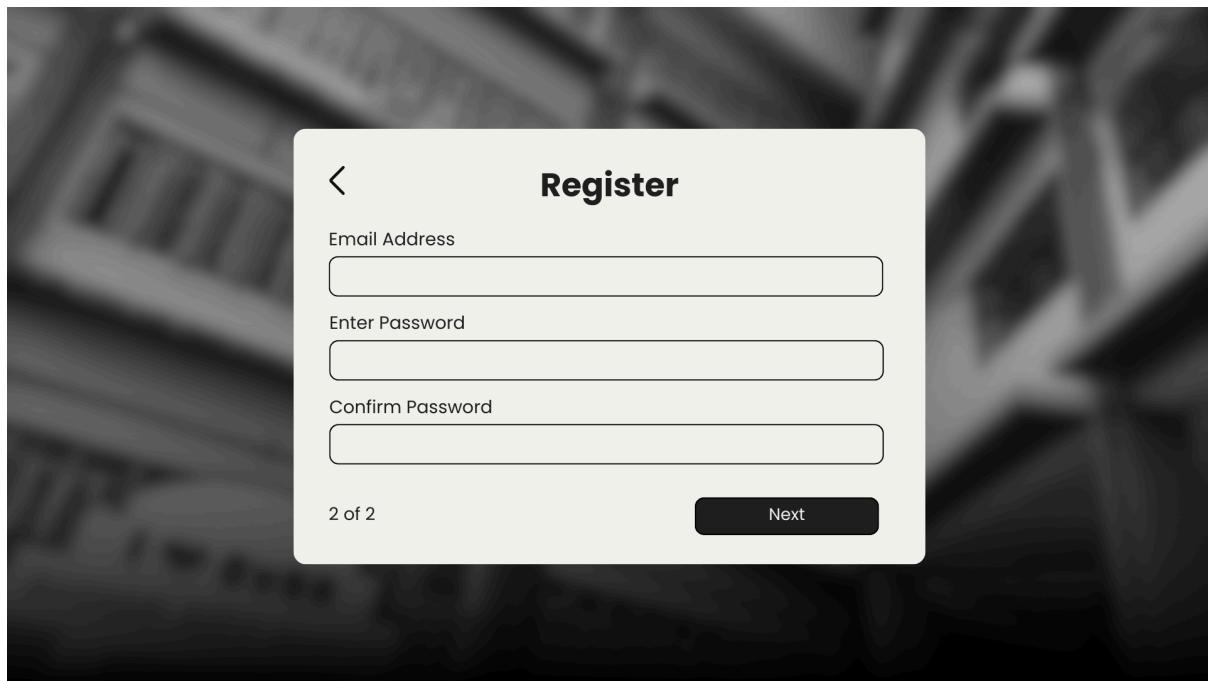
Login and Landing Page



Once the user opens the system, it will show the landing page and the announcements.

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Register, OTP and Success Message



The user must bypass the OTP in order to create an account.



Profile Setup Page and Command Tables

A screenshot of a profile setup page. On the left, there is a placeholder for a user's profile picture with the text "Keith FD" below it. To the right of this are four tabs: "Personal Information" (which is selected and highlighted in black), "Family Background", "Educational Attainment", and "Account Settings". The "Personal Information" section contains fields for First Name, Middle Name, Last Name, Date of Birth (with dropdown menus for MM, DD, and YY), Sex (with radio buttons for Male and Female), Address, City, Province, Postal Code, Country, and Contact Number. At the bottom left of this section is a "View Public Profile" button, and at the bottom right is a "Save Changes" button.

The profile page includes all the personal information of the enrollee.

Enrollment System for Department of Computer Studies – Cavite State University – Bacoor Campus

The screenshot shows the 'Enrollment Team' section of the system. At the top, there are navigation links: Dashboard, Enrollees, Students, Enrollment Team, Log Out, and a bell icon with a red dot. Below the header is a title 'Enrollment Team' and a subtitle 'Enrollment Team List'. A horizontal table header row contains columns for Last Name, First Name, Middle Name, Role, and Commands. Under the 'Commands' column, there are two buttons: 'Delete User' and 'Edit Role'. The main area below the header is currently empty, showing a large white space.

A sample of a table with commands. In this image, the table is for admin's enrollment team management.

The screenshot shows the 'Enrollees' section of the system. At the top, there are navigation links: Dashboard, Enrollees, Students, Enrollment Team, Log Out, and a bell icon with a red dot. Below the header is a title 'Enrollees' and a subtitle 'Computer Science'. A horizontal table header row contains columns for Student ID, Last Name, First Name, Middle Name, Student Type, Year Standing, Status, and Commands. Under the 'Commands' column, there are three buttons: 'Edit Credentials', 'Pre-enrollment Form', and 'Mark as Enrolled'. At the bottom of the page, there are sorting options: 'Sort by: Enrollment Date', 'Student Type: Enrollment Date', 'Year Standing: 5th or more', and a button 'Export as Excel File'.

Located in the admin side of the system for enrollee management.

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Dashboard and Enrollee Status Page

The dashboard displays the following information:

- Enrolled Students:**
 - Total Enrolled: **25**
 - Total Enrolled Computer Science: **25**
 - Total Enrolled Information Technology: **25**
- Regular Students:** 45 (Green bar)
- Irregular Students:** 20 (Red bar)
- Gender:**
 - Information Technology: 20 (Blue bar)
 - Computer Science: 20 (Pink bar)
- Announcement:** A text input field placeholder "Make an announcement..." with a "Send" button.
- Total Paid (IT):** **23**
- Total Paid (CS):** **23**

The dashboard of the enrollment team.

The status and scheduling page shows the following:

- Enrollment Status:** A horizontal progress bar with 5 steps:
 - Step 1: Submission of Necessary Credentials
 - Step 2: Advising and Face-to-Face Submission of Requirements
 - Step 3: Schedule for Enrollment
 - Step 4: Face-to-Face Enrollment
 - Step 5: Fully Enrolled
- Enrollment Date:** **September 20, 2024**
- Rescheduling Feedback:** Princess Mae Binalagbag (APPROVED)

Status page of the enrollees.

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Subjects Selection, Certificate of Recognition, Society Fee Receipt and Pre-Enrollment Form

The screenshot shows a dark-themed web application. At the top, there are navigation links: 'Dashboard', 'Submissions', 'Advisee', 'Log Out', and a notification bell icon with a red dot. Below this, the word 'Advisee' is displayed in large letters. A search bar is present. A central modal window titled 'Edit Subjects' contains a table for managing subjects. The table has two rows. Each row contains a dropdown menu ('Mathematics in the Modern World – GNED 03'), a text input ('3.00'), and two buttons ('Add' and 'Delete'). Below the modal, there are dropdown menus for 'Sort by: Enrollment Date', 'Student Type: S6', and 'Year Standing: 5th or more'.

For advisers, it allows them to edit the subjects of the advisee/enrollee.

The screenshot shows a certificate of enrollment from Cavite State University, Bacoor City Campus. The header includes the university's logo, name, address, and academic year (2023-2024). The main section is titled 'REGISTRATION FORM'. It lists student information: Student Number (202211456), Student Name (DELA CRUZ, JUAN CRUZ), Program (INFORMATION TECHNOLOGY), and Address (#12 Yellow St., Malakong Subdivision, Molino II, Bacoor). It also lists academic details: Semester (1st Semester), Academic Year (2023-2024), Encoder (via DCS Enrollment System), and a note about the validity of the certificate. A table shows course registration: PSY50 (2 LEC, 1 LAB, COURSE TITLE: PHILOSOPHY IN AMNESIA). At the bottom, there is a registration status section and a signature area.

CLASS CODE	UNITS	COURSE TITLE
PSY50	2 LEC 1 LAB	PHILOSOPHY IN AMNESIA
TOTAL: 20		

*** NOTHING FOLLOWS ***

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

	<p style="text-align: center;"> Republic of the Philippines CAVITE STATE UNIVERSITY BACOOR CITY CAMPUS</p> <p style="text-align: center;">PRE-ENROLLMENT FORM</p> <table border="1" style="width: 100%; border-collapse: collapse;"><tr><td colspan="2" style="padding: 5px;">Personal Information</td></tr><tr><td style="width: 50%;">Name Address Program Student Category</td><td style="width: 50%;">Email Address Contact Number</td></tr><tr><td colspan="2" style="padding: 5px;">Subjects</td></tr><tr><td style="width: 33%;">Class Code</td><td style="width: 33%;">Course/Subject Title</td><td style="width: 33%;">Number of Units</td></tr><tr><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td></tr></table>	Personal Information		Name Address Program Student Category	Email Address Contact Number	Subjects		Class Code	Course/Subject Title	Number of Units																																	
Personal Information																																											
Name Address Program Student Category	Email Address Contact Number																																										
Subjects																																											
Class Code	Course/Subject Title	Number of Units																																									

	  <p>CAVITE STATE UNIVERSITY BACOOR CITY CAMPUS Department of Computer Studies SOCIETY FEE RECEIPT</p> <table border="1" style="width: 100%; border-collapse: collapse;"><tr><td style="width: 25%;">Student Number</td><td style="width: 25%;">202211456</td><td style="width: 25%;">Amount Paid</td><td style="width: 25%;">100.00 PESOS ONLY</td></tr><tr><td>Student Name</td><td>DELA CRUZ, JUAN CRUZ</td><td>Academic Year</td><td>2023-2024</td></tr><tr><td>Program</td><td>INFORMATION TECHNOLOGY</td><td>Semester</td><td>1st Semester</td></tr><tr><td>Received by</td><td>Claire Ferrer</td><td colspan="2"></td></tr><tr><td>Signature</td><td>-----</td><td colspan="2"></td></tr></table>	Student Number	202211456	Amount Paid	100.00 PESOS ONLY	Student Name	DELA CRUZ, JUAN CRUZ	Academic Year	2023-2024	Program	INFORMATION TECHNOLOGY	Semester	1st Semester	Received by	Claire Ferrer			Signature	-----			
Student Number	202211456	Amount Paid	100.00 PESOS ONLY																			
Student Name	DELA CRUZ, JUAN CRUZ	Academic Year	2023-2024																			
Program	INFORMATION TECHNOLOGY	Semester	1st Semester																			
Received by	Claire Ferrer																					
Signature	-----																					

Certificate of Registration, Pre-Enrollment Form, and Society Fee Receipts are printable once the student is done with the specified processes assigned to it.

Frequently Asked Questions and Enrollment Guide



Admission Process

Please read the process before proceeding to the next steps of enrollment.

Students are categorized by the following:

- Regular Students (S1): Those who are taking the regular load of subjects.
- Incoming First-Year Students (S2).
- Irregular Students (S3): Those who are not taking the expected load of subjects based on their year level.
- Shiftees (S4): Students who have shifted from other programs to IT/CS.
- Returnees (S5).
- Transferees (S6): Students who have transferred from other universities.

Conditions for the classification of student types are outlined as follows:

1. A student who is a transferee and/or a regular student from other Cvsu campuses shall be classified as S6.
2. The classification of S4 shall be applied exclusively to students residing in Cvsu Bacoor who shift to the IT/CS program.
3. A student shall be classified as S3 only if they are residing on campus and are not enrolled in the regular workload of subjects.
4. Students will be automatically classified as S3 if their Certificate of Grades indicates a score of 4.00, 5.00, or contains the designations INC or DRP.
5. Students that are planning to shift to other programs and/or transfer to other campuses shall not use this system.

Requirements

For Regular Students (S1) and Irregular Students (S3), the following requirements should be submitted in the system for upon checking:

1. Scanned Copy of Curriculum Checklist signed by advisers last semester; and
2. Last semester's Certificate of Registration.

For Transferee (S6):

1. Transcript of Records (TOR)

Incoming First Year Students (S2) will not submit any requirements on the system as it submissions are arranged by the registrar.

Shiftees (S4) and Returnees (S5) will also not submit anything via online, but it is required to pick an advising schedule to submit all the requirements needed via face-to-face.

After verification and checking by the advisers: **S1 and S2** will be given an **Enrollment Date**, while **S4, S5, and S6** will be given a chance to select an advising date for picking schedules, sections and subjects.

Steps of Enrollment

For Regular Students (S1 and S2):

1. Submit the necessary scanned documents through the system.
2. Once submitted, you will receive your enrollment date.
3. On your enrollment date, pay the society fee and wait for the issuance of your Certificate of Registration.

For Irregular Students (S3, S4, S5, and S6):

1. Submit the necessary scanned documents through the system.
2. Select an advising date for the submission of requirements and the selection of subjects.
3. Once completed, you will receive your enrollment date.
4. On your enrollment date, pay the society fee and wait for the issuance of your Certificate of Registration.

Frequently Asked Questions

+ How can I change my password?
+ How and where can I pay matriculation and society fees?
+ Should I still use the system if I'm planning to shift to other programs outside DCS?
+ Why I can't select an advising date that I want?

I have read and understand the enrollment process.

Proceed

IMPLEMENTATION

DEVELOPMENT TOOLS AND ENVIRONMENT

- **Development Tools**

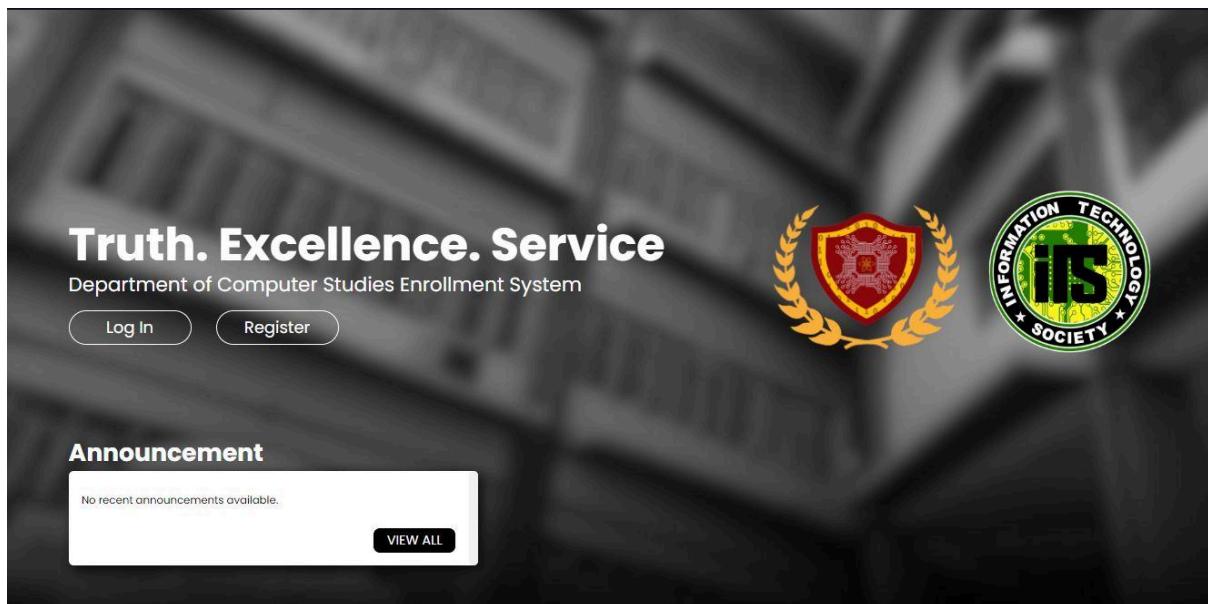
- IDE/Code Editor: Visual Studio Code
- Documentation and Progress Reports: Google Docs
- Version Control: Git and Github
- Team Meetings and Collaborations: Google Meet
- Diagrams: Creately and Draw.io
- UI/UX Design: Figma and Canva
- Management and Tracking: Physical Kanban Board and Trello
- Database Management: MySQL Workbench

- **Programming Languages and Framework**

- Programming Language: *JavaScript*
- User Interface: *CSS3 and HTML5*
- Front-End Development Framework: *React.js*
- Back-End Development Framework: *Node.js with Express.js*
- Testing Front-End Components: *Jest + React Testing Library*
- Testing Back-End Components: *Jest with SuperTest*

FEATURES AND FUNCTIONALITIES

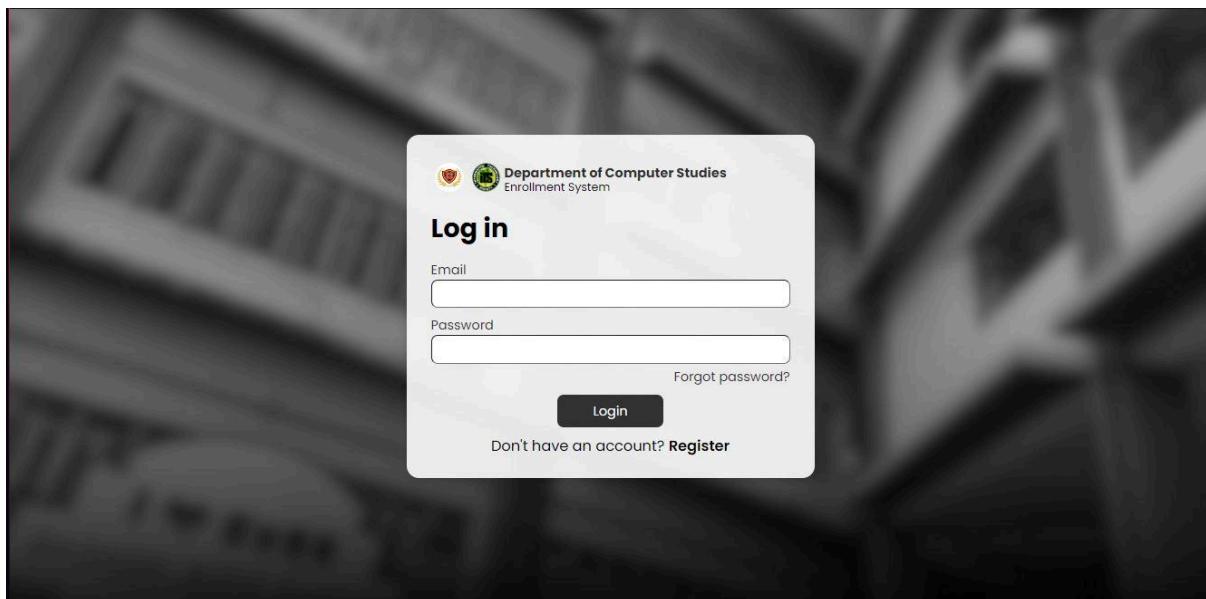
Landing and Announcements Page



This screenshot shows the "Announcements" section. It features a header with a black dot icon and the word "Announcements". Below the header, a message states "No announcements available." and includes a "VIEW ALL" button. The main body of the section is currently empty.

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Login, Register and Password Reset Pages



System Administrator: Dashboard, Student, Enrollee and Table Commands

The dashboard has a top navigation bar with icons for the university logo, Dashboard, Enrollees, Students, Enrollment Team, Log Out, and a bell icon. The main area is titled "Dashboard". It contains several data cards:

- Enrolled Students**:
 - Total Enrolled: 1
 - Total Enrolled Computer Science: 1
 - Total Enrolled Information Technology: 0
- Gender**:
 - Regular Student: 45 (green)
 - Irregular Student: 20 (red)
- Announcement**: A text input field with placeholder "Make an announcement..."
- Total Paid (IT)**: 0
- Total Paid (cs)**: 0

***Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus***


[Dashboard](#)
[Enrollees](#)
[Students](#)
[Enrollment Team](#)
[Log Out](#)
🔔

Student

Student ID	Last Name	First Name	Middle Name	Student Type	Year Standing	Commands
20231001	Labalan	Jerald	Skip	S6	3rd year	<button style="background-color: orange; color: white; border: none; padding: 2px 5px;">Print COR</button>
20231002	Fernandez	Alex	Hawak	S6	3rd year	<button style="background-color: orange; color: white; border: none; padding: 2px 5px;">Print COR</button>
20231003	Galvez	Dioren	Golem	S6	3rd year	<button style="background-color: orange; color: white; border: none; padding: 2px 5px;">Print COR</button>
20231004	Dasalla	Keith	Skip	S5	3rd year	<button style="background-color: orange; color: white; border: none; padding: 2px 5px;">Print COR</button>
20231005	Bides	Matthew	Tigastite	S5	2nd year	<button style="background-color: orange; color: white; border: none; padding: 2px 5px;">Print COR</button>

 Sort by: Student ID

 Student Type: All

 Year Standing: All

Export as Spreadsheet


[Dashboard](#)
[Enrollees](#)
[Students](#)
[Enrollment Team](#)
[Log Out](#)
🔔

Enrollees

Student ID	Last Name	First Name	Program	Student Type	Year Standing	Commands

 Program: All

 Student Type: All

 Year Standing: All

Export as Spreadsheet


[Dashboard](#)
[Enrollees](#)
[Students](#)
[Enrollment Team](#)
[Log Out](#)
🔔

Enrollment Team

+

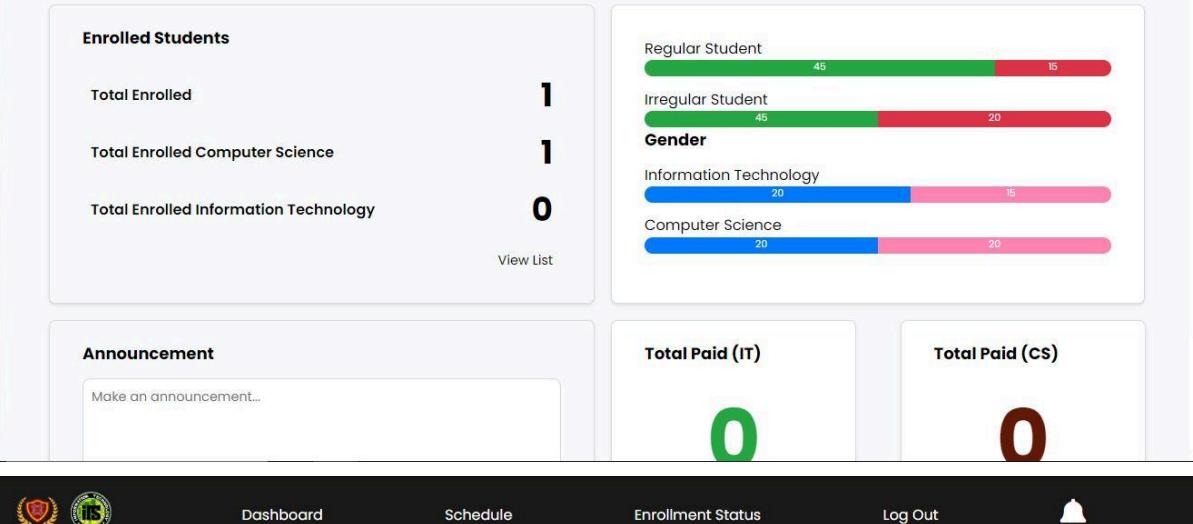
First Name	Middle Name	Last Name	Role	Commands
Jerald	Valdez	Labalan	Kopal	<button style="background-color: orange; color: white; border: none; padding: 2px 5px;">Edit Role</button> <button style="background-color: red; color: white; border: none; padding: 2px 5px;">Delete</button>

localhost:3000/min

Enrollment System for Department of Computer Studies – Cavite State University – Bacoor Campus

Society Officers: Dashboard, Change Enrollment Status and Schedule Enrollment Table Commands

Dashboard



Enrollment Status

Search...

Student ID	Last Name	First Name	Student Type	Year Standing	Status	Commands
20231001	Labalan	Jerald	S6	3rd year	Step 1: Done	<button>Mark as Enrolled</button> <button>Print Receipt</button>
20231002	Fernandez	Alex	S6	3rd year	Step 1: Done	<button>Mark as Enrolled</button> <button>Print Receipt</button>
20231003	Galvez	Dioren	S6	3rd year	Step 1: Done	<button>Mark as Enrolled</button> <button>Print Receipt</button>
20231004	Dasalla	Keith	S5	3rd year	Step 1: Done	<button>Mark as Enrolled</button> <button>Print Receipt</button>
20231005	Bides	Matthew	S5	2nd year	Step 1: Done	<button>Mark as Enrolled</button> <button>Print Receipt</button>

Student Type: All

Year Standing: All

[Export as Spreadsheet](#)



Schedule

Search...

Student ID	Last Name	First Name	Student Type	Year Standing	Advising Date	Commands
20231001	Labalan	Jerald	S6	3rd year	11/11/11	<button>Accept Schedule</button> <button>Redesignate Schedule</button>
20231002	Fernandez	Alex	S6	3rd year	11/11/11	<button>Accept Schedule</button> <button>Redesignate Schedule</button>
20231003	Galvez	Dioren	S6	3rd year	11/11/11	<button>Accept Schedule</button> <button>Redesignate Schedule</button>
20231004	Dasalla	Keith	S5	3rd year	11/11/11	<button>Accept Schedule</button> <button>Redesignate Schedule</button>
20231005	Bides	Matthew	S5	2nd year	11/11/11	<button>Accept Schedule</button> <button>Redesignate Schedule</button>

Schedule Date: All

Student Type: All

Year Standing: All

localhost:3000

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

Adviser: Dashboard, View Enrollees' File Submissions and Change Proposed Subjects

Dashboard

Enrolled Students

Total Enrolled

1

Total Enrolled Computer Science

1

Total Enrolled Information Technology

0

[View List](#)

Regular Student

45

15

Irregular Student

45

20

Gender

Information Technology

20

15

Computer Science

20

20

Announcement

Make an announcement...

Total Paid (IT)

0

Total Paid (CS)

0



Dashboard

Submissions

Advisee

Log Out



Submissions

Search...

Student ID	Last Name	First Name	Middle Name	Student Type	Year Standing	Commands			
20231001	Labalan	Jerald	Sikip	S6	3rd year	<button>View Checklist</button>	<button>View ISCOR</button>	<button>Reject</button>	<button>Accept</button>
20231002	Fernandez	Alex	Hawak	S6	3rd year	<button>View Checklist</button>	<button>View ISCOR</button>	<button>Reject</button>	<button>Accept</button>
20231003	Galvez	Dioren	Golem	S6	3rd year	<button>View Checklist</button>	<button>View ISCOR</button>	<button>Reject</button>	<button>Accept</button>
20231004	Dasalla	Keith	Sikip	S5	3rd year	<button>View Checklist</button>	<button>View ISCOR</button>	<button>Reject</button>	<button>Accept</button>
20231005	Bides	Matthew	Tigastite	S5	2nd year	<button>View Checklist</button>	<button>View ISCOR</button>	<button>Reject</button>	<button>Accept</button>

Sort by:

Student Type:

Year Standing:

localhost:3000

***Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus***

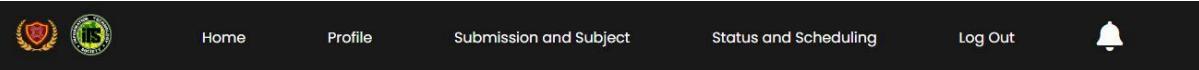
Enrollee: Home, Profile Set-up, Enrollment/Admission Guide, FAQS, Subjects and Submissions, Status and

Scheduling

The screenshot shows the home page of the enrollment system. At the top, there are two circular icons: one with a shield and another with a book. The navigation menu includes Home, Profile, Submission and Subject, Status and Scheduling, Log Out, and a bell icon. The main content area has a large gray placeholder for a profile picture. Below it, the user's information is listed: Name (Bides, Matt Isaac), Student Number (N/A), Program (N/A), and Year Level (N/A). A "View Details" button is at the bottom. To the right, the enrollment status is displayed as "Not Enrolled" in large red text, with a yellow banner below it stating "No Enrollment Schedule Date". Buttons for "Print COR" and "Print COE" are also present. Below this, there is a section for announcements titled "Society President" with a message in Tagalog: "Sa mga students na late mag-enroll.. Paano kaya maggrow nyan, hindi kayo naghihiwalay." A "VIEW ALL" button is at the bottom right.

The screenshot shows the profile set-up page. On the left, there is a large circular placeholder for a profile picture with a camera icon and a "Profile" label. On the right, there is a form divided into four tabs: Personal Information (selected), Family Background, Educational Attainment, and Account Settings. Under Personal Information, fields include First Name, Middle Name, Last Name, and Suffix, each with an input field. Below these are Date of Birth (mm/dd/yyyy), Sex (radio buttons for Male and Female), House No., Barangay, City, Province, Postal Code, and Country. A Contact Number field with an input field is also present. At the bottom right of the form is a "Save Changes" button.

Enrollment System for Department of Computer Studies – Cavite State University – Bacoor Campus



Subjects and Submission

Student Categorization

S1 - Regular students
S2 - Incoming First Year Student
S3 - Irregulars
S4 - Shiftee
S5 - Returnee
S6 - Transferee

Student ID

Year Level

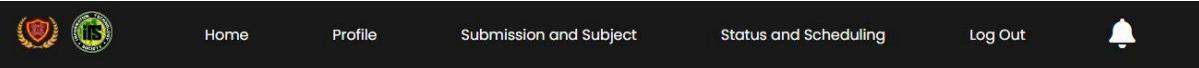
Program

Select Student Type

You may refer to the categories above.

S1 S2 S3 S4 S5 S6

Next



Status and Scheduling

Enrollment Status



Enrollment Date

September 20, 2024

Appeal for Reschedule

Rescheduling Feedback

Princess Mae Binalagbag

APPROVED



Admission Process

Please read the process before proceeding to the next steps of enrollment. Students are categorized by the following:

Students are categorized by the following:

- Regular Students (S1): Those who are taking the regular load of subjects.
- Incoming First-Year Students (S2):
- Irregular Students (S3): Those not taking the expected load based on their year level.
- Shiftees (S4): Students who have shifted from other programs to IT/CS.
- Returnees (S5):
- Transferees (S6): Students who transferred from other universities.

Conditions for the classification of student types are outlined as follows:

1. A student who is a transferee and/or a regular student from other CvSU campuses shall be classified as S6
2. The classification of S4 shall be applied exclusively to students residing in CvSU Bacoor who shift to the IT/CS program.
3. A student shall be classified as S3 only if they are residing on campus and are not enrolled in the regular workload of subjects.
4. Students will be automatically classified as S3 if their Certificate of Grades indicates a score of 4.00, 5.00, or contains the designations INC or DRP.
5. Students that are planning to shift to other programs and/or transfer to other campuses shall not use this system.

Requirements For Regular Students (S1) and Irregular Students (S3).



Home Profile Submission and Subject Status and Scheduling Log Out 

Subjects and Submission

Student Categorization

S1 – Regular students
S2 – Incoming First Year Student
S3 – Irregulars
S4 – Shiftee
S5 – Returnee
S6 – Transferee

Student ID

Year Level

Program

Select Student Type
You may refer to the categories above.
 S1 S2 S3 S4 S5 S6

Next

TESTING

TEST PLAN

The general purpose of this test plan is to evaluate the reception of the enrollment system, ensuring that it meets established quality standards through comprehensive testing across various dimensions. The scope of testing includes:

Functional Testing, verifies that all system features operate as expected and fulfill the specified requirements.

Usability Testing assesses the user interface and overall user experience, ensuring that the system is intuitive and accessible for all users.

Performance Testing evaluates the system's responsiveness and stability under different load conditions, ensuring it can handle peak usage without degradation in performance.

Unit Testing focuses on individual components of the system to confirm that each part functions correctly in isolation.

Security Testing identifies vulnerabilities and potential threats within the system, ensuring that sensitive data is protected

Acceptance Testing involves end-users to validate that the system meets their needs and expectations before final deployment. Together, these testing types will provide a thorough assessment of the enrollment system's quality and readiness for production.

TEST CASES

Action	Activities	System Response	Expected Error	System Response
Login	Type password and username -Login successfully	Check all fields -Verify input -Authentication to database	Incorrect password and Username	Prompt incorrect password and username

OTP Verification	Input the email of your account	Check all fields -Save to database	None	None
Change Password	Password confirmation are not the same	Compare two inputs	Passwords does not match	Repeat user process
Register Account	Type Email and Password -Create account successfully	Check all fields -Verify input -Save to database	None	None
Add Announcements	Specify all fields -Type the announcement message	Check all fields -Save to database	None	None

UNIT TESTING

Below are the screenshots of each test file that passed for the UI components and functionalities.

System Administrator

```
PASS  src/Admin/Students/CORPrint.test.js
Print COR
  ✓ renders header section (163 ms)
  ✓ renders student information section with default data (23 ms)
  ✓ renders student information section with mock data (24 ms)
  ✓ renders certification section (16 ms)
  ✓ renders course table (20 ms)
  ✓ renders footer section (24 ms)

Test Suites: 1 passed, 1 total
Tests:       6 passed, 6 total
Snapshots:  0 total
Time:        8.471 s
```

Print COR option

```
PASS  src/Admin/Dashboard/Dashboard.test.js
Dashboard Component
  ✓ renders loading state (126 ms)
  ✓ renders error state (12 ms)
  ✓ renders enrolled students with counts (15 ms)
  ✓ handles announcement input (15 ms)
  ✓ renders total paid values (11 ms)

Test Suites: 1 passed, 1 total
Tests:      5 passed, 5 total
Snapshots:  0 total
Time:       8.712 s
```

Dashboard for the enrollment team

```
PASS  src/Admin/Dashboard/DashboardHeader.test.js
Header Component
  ✓ renders logos correctly (304 ms)
  ✓ renders all navigation links text correctly (33 ms)
  ✓ calls handleLogout on 'Log Out' click (72 ms)
  ✓ renders notification icon (8 ms)
  ✓ navigation links route correctly (10 ms)
  ✓ renders clickable logos with correct links (8 ms)
  ✓ Log Out link has correct style applied (47 ms)

Test Suites: 1 passed, 1 total
Tests:      7 passed, 7 total
Snapshots:  0 total
Time:       8.95 s
```

Dashboard header

```
PASS  src/Admin/EnrollmentStatus/HeaderNavbar.test.jsx (7.052 s)
HeaderNavbar Component
  ✓ renders the app bar with the correct background color (190 ms)
  ✓ renders the logos correctly (10 ms)
  ✓ renders all navigation buttons (78 ms)
  ✓ renders the layout with correct structure (9 ms)

Test Suites: 1 passed, 1 total
Tests:      4 passed, 4 total
Snapshots:  0 total
Time:       11.012 s
```

Navigation Bars

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

```
PASS | src/Admin/EnrollmentTeam/EnrollmentTeam.test.js
Enrollment Team Component
  ✓ renders the Enrollment Team table and filters correctly (166 ms)
  ✓ handles filter selection correctly (22 ms)
Enrollment Team Modal
  ✓ renders the Add User Modal with user information and form fields (33 ms)
  ✓ handles role selection correctly (16 ms)
  ✓ handles form input changes (21 ms)

Test Suites: 1 passed, 1 total
Tests:      5 passed, 5 total
Snapshots:  0 total
Time:       7.788 s
```

Enrollment team table and commands

```
PASS | src/Admin/EnrollmentStatus/EnrollmentStatus.test.jsx
Enrollment Status Component
  ✓ renders search input and handles input changes (138 ms)
  ✓ renders table headers and data rows (37 ms)
  ✓ handles status select changes (27 ms)
  ✓ handles sort by filter changes (21 ms)
  ✓ handles student type and year standing filter changes (29 ms)

Test Suites: 1 passed, 1 total
Tests:      5 passed, 5 total
Snapshots:  0 total
Time:       7.116 s
```

Enrollment status

```
PASS | src/Admin/Enrollees/Enrollees.test.js (5.201 s)
Enrollees Component
  ✓ renders the Enrollees Component (165 ms)
  ✓ calls handleSearchChange on search input change (18 ms)
  ✓ calls handleRejectClick when clicking 'Mark as Enrolled' (18 ms)
  ✓ calls handleFilterProgramChange on filter program selection (21 ms)
  ✓ renders the table with student data (45 ms)
  modal for view checklist and reject
    ✓ renders the View Checklist Modal (8 ms)
    ✓ renders the Reject Modal (12 ms)

Test Suites: 1 passed, 1 total
Tests:      7 passed, 7 total
Snapshots:  0 total
Time:       9.556 s
```

Enrollee table and commands

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

```
PASS  src/Admin/Students/Students.test.js
Student Table Component
  ✓ renders the component with table headers (282 ms)
  ✓ filters students by search query (26 ms)
  ✓ sorts students by last name (58 ms)
  ✓ calls handlePrintCOR when Print COR button is clicked for a specific student (17 ms)
  ✓ calls handleExportSpreadsheet when export button is clicked (17 ms)

Test Suites: 1 passed, 1 total
Tests:      5 passed, 5 total
Snapshots:  0 total
Time:       6.657 s
```

Admin's student table

```
PASS  src/Admin/SetEnrollment/SetEnrollment.test.js
Set Enrollment Component
  ✓ renders welcome message and buttons (109 ms)
  ✓ renders modal when 'set enrollment date' button is clicked (14 ms)
  ✓ displays modal content with the correct step (18 ms)
  ✓ renders step-specific modal content (15 ms)
  ✓ closes modal when clicking outside or on the close button (12 ms)
  ✓ renders success message in step 4 (7 ms)
Set Enrollment Modal
  ✓ renders modal and next button on step 1 (4 ms)
  ✓ renders step 2 with back and next buttons (7 ms)
  ✓ renders step 3 with back and submit buttons (9 ms)
  ✓ renders step 4 with go to dashboard button (3 ms)
  ✓ calls nextStep on next button click (6 ms)
  ✓ calls prevStep on back button click (7 ms)

Test Suites: 1 passed, 1 total
Tests:      12 passed, 12 total
Snapshots:  0 total
Time:       7.845 s
```

Set enrollment period

```
Modal functionality
  ✓ opens the modal when the 'Add User' button is clicked (77 ms)
  ✓ closes the modal when the close button is clicked (24 ms)

Test Suites: 1 passed, 1 total
Tests:      5 passed, 5 total
Snapshots:  0 total
Time:       2.752 s
Ran all test suites matching /Admin\\EnrollmentTeam\\EnrollmentTeam.test.js/i.

Watch Usage: Press w to show more.[]
```

Modal for adding a member in enrollment team

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS

```
✓ renders correctly (12 ms)
✓ opens and closes the modal (10 ms)
✓ advances through modal steps (28 ms)

Test Suites: 1 passed, 1 total
Tests:      3 passed, 3 total
Snapshots:  0 total
Time:       2.479 s
Ran all test suites matching /Admin\\SetEnrollment\\SetEnrollment.test.js/i.

Watch Usage: Press w to show more.[]
```

Set enrollment period modal

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS

```
✓ shows all students by default (62 ms)
✓ searches students by name (27 ms)

Test Suites: 1 passed, 1 total
Tests:      3 passed, 3 total
Snapshots:  0 total
Time:       3.143 s
Ran all test suites matching /Admin\\Students\\Students.test.js/i.

Watch Usage: Press w to show more.[]
```

Admin's students table and commands

Adviser

```
PASS  src/Adviser/Header/Header.test.js
Header Component
  ✓ renders the logo images (285 ms)
  ✓ renders all navigation links with correct text (30 ms)
  ✓ renders all nav links without errors (110 ms)

Test Suites: 1 passed, 1 total
Tests:      3 passed, 3 total
Snapshots:  0 total
Time:       5.317 s
```

Header

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

```
PASS  src/Adviser/A_Dashboard/Dashboard.test.js
Dashboard Component
  ✓ renders all cards and titles correctly (277 ms)
  ✓ renders the Pie chart (17 ms)
  ✓ renders progress bars with correct styles (44 ms)
  ✓ handles announcement input and button click (24 ms)
  ✓ renders Total Paid (IT) and Total Paid (CS) cards (21 ms)

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:   0 total
Time:        5.872 s
```

Dashboard

```
PASS  src/Adviser/Advisee/Advisee.test.js
Advisee Component
  ✓ renders the title and search input (186 ms)
  ✓ handles search input change (36 ms)
  ✓ renders the table headers (193 ms)
  ✓ renders student rows with commands (75 ms)
  ✓ triggers Edit Subject command (26 ms)
  ✓ renders and handles sorting dropdown (26 ms)
  ✓ renders and handles Student Type filter dropdown (20 ms)
  ✓ renders and handles Year Standing filter dropdown (13 ms)
```

Advisee/set advising date

```
PASS  src/Adviser/A_Submission/Submission.test.js
SubmissionsComponent
  ✓ renders header and title (241 ms)
  ✓ renders search input (23 ms)
  ✓ renders table and student rows (21 ms)
  ✓ triggers action buttons (22 ms)
  ✓ renders filter bar and updates filters (32 ms)

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:   0 total
Time:        6.852 s
```

File submissions

All User Sides

```
PASS src/General/Login/Login.test.js
Login Component
  ✓ renders login form fields and button (272 ms)
  ✓ shows "Invalid email address" message for wrong email (115 ms)
  ✓ shows "Invalid password" message for wrong password (49 ms)

Test Suites: 1 passed, 1 total
Tests:       3 passed, 3 total
Snapshots:   0 total
Time:        6.962 s
```

Login

```
PASS src/General/LandingPage/Landing.test.js
LandingPage Component
  ✓ renders the main structure and text correctly (456 ms)
  ✓ renders login and register buttons (154 ms)
  ✓ displays a message when no announcements are available (15 ms)
  ✓ includes a link to view all announcements (50 ms)

Test Suites: 1 passed, 1 total
Tests:       4 passed, 4 total
Snapshots:   0 total
Time:        7.563 s
```

Landing

```
PASS src/General/Announcement/AnnouncementsPage.test.js
AnnouncementsPage Component
  ✓ displays a message when no announcements are available (310 ms)
  ✓ handles errors during API call gracefully (38 ms)

Test Suites: 1 passed, 1 total
Tests:       2 passed, 2 total
Snapshots:   0 total
Time:        6.585 s
```

Announcements

Society Officers

```
PASS src/SocietyOfficer/EnrollmentStatus/EnrollmentStatusCS.test.js
EnrollmentStatusCS Component
  ✓ renders the header text (487 ms)
  ✓ renders search input with placeholder (30 ms)
  ✓ renders table headers (63 ms)
  ✓ renders filter labels and options (105 ms)
  ✓ renders buttons in the table (21 ms)
  ✓ renders the export button (19 ms)
  ✓ renders view modal with student details (36 ms)
  ✓ renders the reject modal and handles user interactions (34 ms)

Test Suites: 1 passed, 1 total
Tests: 8 passed, 8 total
Snapshots: 0 total
Time: 6.894 s
```

Enrollment Status

```
PASS src/SocietyOfficer/Header/HeaderCS.test.js
HeaderCS Component
  ✓ renders the header with logos (279 ms)
  ✓ renders navigation links with correct text (19 ms)
  ✓ renders the notification bell icon (94 ms)

Test Suites: 1 passed, 1 total
Tests: 3 passed, 3 total
Snapshots: 0 total
Time: 6.393 s
```

Header

```
PASS src/SocietyOfficer/EnrollmentStatus/ReceiptPrint.test.js
Receipt Component
  ✓ renders header section correctly (255 ms)
  ✓ renders logos with correct alt text (22 ms)
  ✓ renders receipt details correctly (52 ms)
  ✓ renders placeholder values when student data is missing (62 ms)
  ✓ renders signature line (33 ms)

Test Suites: 1 passed, 1 total
Tests: 5 passed, 5 total
Snapshots: 0 total
Time: 6.216 s
```

Print receipt

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

```
PASS  src/SocietyOfficer/Schedule/ScheduleCS.test.js
Schedule Component
  ✓ renders the header correctly (225 ms)
  ✓ renders the title correctly (23 ms)
  ✓ renders the search input (21 ms)
  ✓ filters students based on search query (39 ms)
  ✓ displays the students table with the correct headers (37 ms)
  ✓ clicking the 'Accept Schedule' button (19 ms)
  ✓ clicking the 'Redesignate Schedule' button (24 ms)
  ✓ filters students by schedule date (37 ms)
  ✓ filters students by student type (22 ms)
  ✓ filters students by year standing (20 ms)

Test Suites: 1 passed, 1 total
Tests:      10 passed, 10 total
Snapshots:  0 total
Time:       5.837 s
```

Schedule enrollment date

```
21 | test/“Ensures the table with data” | ✓ - ✘
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

EnrollmentStatus Component
  ✓ renders the component title (53 ms)
  ✓ renders the search input and updates its value (18 ms)
  ✓ renders the table with data (42 ms)
  ✓ renders and updates the status dropdown (47 ms)
  ✓ renders 'Mark as Enrolled' buttons (24 ms)

Test Suites: 1 passed, 1 total
Tests:      5 passed, 5 total
Snapshots:  0 total
Time:       2.263 s
Ran all test suites matching /Admin\\EnrollmentStatus\\EnrollmentStatus.test.js/i.

Watch Usage: Press w to show more.[]
```

Enrollment status

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

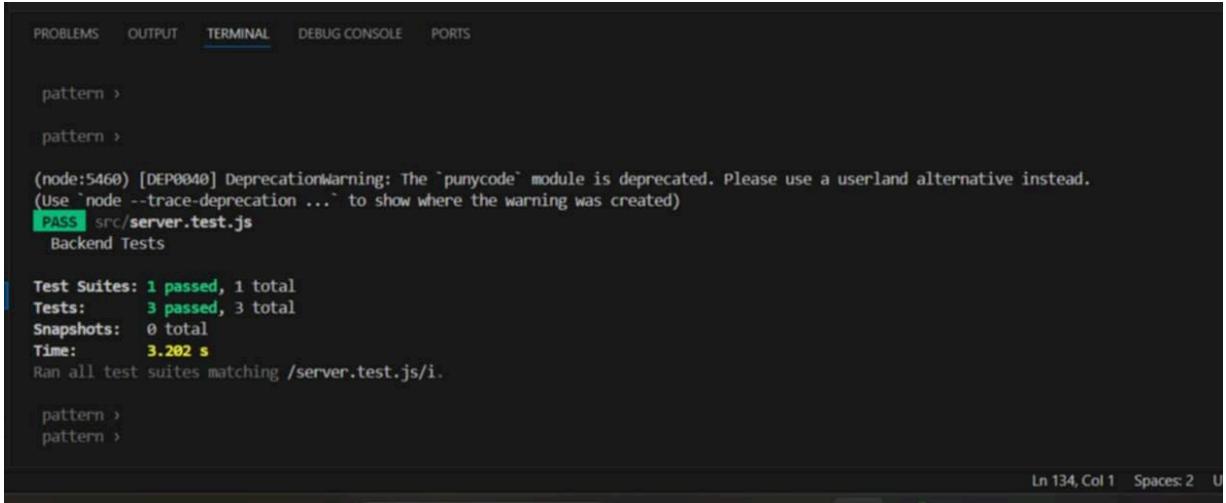
  ✓ renders the announcement textarea and button (115 ms)
  ✓ renders the total paid values (15 ms)
  ✓ handles the 'View List' link click (19 ms)

Test Suites: 1 passed, 1 total
Tests:      4 passed, 4 total
Snapshots:  0 total
Time:       6.299 s
Ran all test suites matching /Adviser\\A_Dashboard\\Dashboard.test.js/i.

Watch Usage: Press w to show more.[]
```

Dashboard

Enrollee



A screenshot of a terminal window from a code editor. The tab bar at the top shows 'PROBLEMS', 'OUTPUT', 'TERMINAL' (which is underlined), 'DEBUG CONSOLE', and 'PORTS'. The terminal output shows the following:

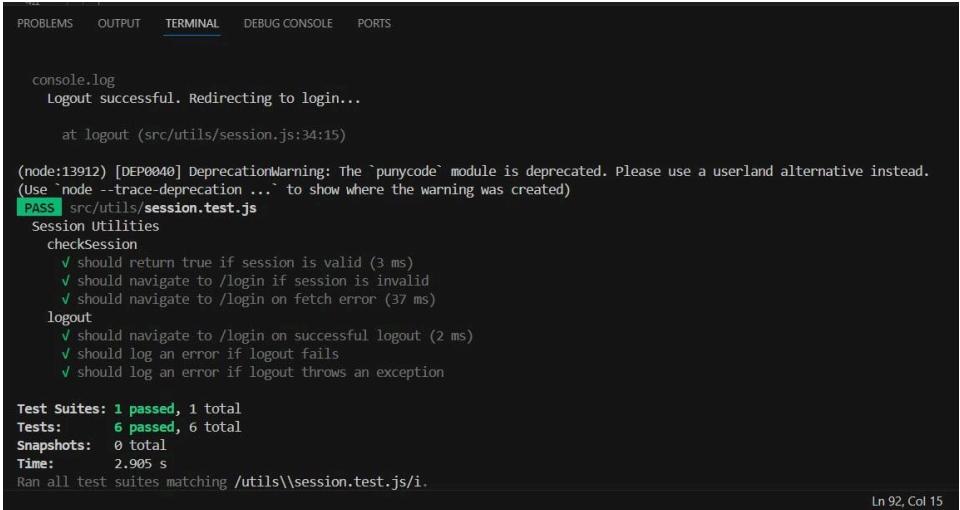
```
pattern >
pattern >
(node:5460) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
PASS | src/server.test.js
Backend Tests

Test Suites: 1 passed, 1 total
Tests:       3 passed, 3 total
Snapshots:  0 total
Time:        3.202 s
Ran all test suites matching /server.test.js/i.

pattern >
pattern >
```

The status bar at the bottom right indicates "Ln 134, Col 1 Spaces: 2 U".

Backend testing



A screenshot of a terminal window from a code editor. The tab bar at the top shows 'PROBLEMS', 'OUTPUT', 'TERMINAL' (underlined), 'DEBUG CONSOLE', and 'PORTS'. The terminal output shows the following:

```
console.log
Logout successful. Redirecting to login...
at logout (src/utils/session.js:34:15)

(node:13912) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
PASS | src/utils/session.test.js
session Utilities
checkSession
  ✓ should return true if session is valid (3 ms)
  ✓ should navigate to /login if session is invalid
  ✓ should navigate to /login on fetch error (37 ms)
logout
  ✓ should navigate to /login on successful logout (2 ms)
  ✓ should log an error if logout fails
  ✓ should log an error if logout throws an exception

Test Suites: 1 passed, 1 total
Tests:       6 passed, 6 total
Snapshots:  0 total
Time:        2.905 s
Ran all test suites matching /utils\\session.test.js/i.
```

The status bar at the bottom right indicates "Ln 92, Col 15".

Session

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

```
PASS  src/Student/AdmissionAndFaqs/AdmissionAndFaqs.test.js
AdmissionAndFaqs Component
  ✓ renders the Navbar with logos (125 ms)
  ✓ renders the Admission Process heading (17 ms)
  ✓ renders requirements for student types (18 ms)
  ✓ renders student categories list (9 ms)
  ✓ renders classification conditions list (25 ms)
  ✓ renders enrollment steps for Regular Students (S1 and S2) (24 ms)
  ✓ renders enrollment steps for Irregular Students (S3, S4, S5, and S6) (18 ms)
  ✓ renders advisory instructions for Incoming First Year Students (S2) (9 ms)
  ✓ renders advisory instructions for Shiftees (S4) and Returnees (S5) (10 ms)

FAQ Section
  ✓ renders FAQ questions and toggles answers (16 ms)
  ✓ renders confirmation checkbox and button (12 ms)

Test Suites: 1 passed, 1 total
Tests:       11 passed, 11 total
Snapshots:   0 total
Time:        6.226 s
```

Enrollment guide and FAQS

```
PASS  src/Student/EmailVerification>EmailVerification.test.js
Email Verification Component
  Step 1: Email Input
    ✓ renders email input and 'Send Verification Code' button (104 ms)
    ✓ captures email input value correctly (16 ms)
    ✓ clicking 'Send Verification Code' triggers handleEmailSubmit (64 ms)

  Step 2: OTP Input
    ✓ renders verification step when step changes to 2 (50 ms)
    ✓ triggers handleVerify when OTP is submitted (15 ms)

  Step 3: Set New Password Component
    ✓ renders step 3 of the Set New Password flow correctly (13 ms)
    ✓ handles input for new password and confirm password fields (12 ms)
    ✓ displays an error message if passwordError is provided (5 ms)
    ✓ calls handlePasswordChange when the Change Password button is clicked (4 ms)

  Step 4 - Success Message Component
    ✓ renders success message and button correctly (6 ms)
    ✓ navigates to login page when "Go to Login" button is clicked (8 ms)

Test Suites: 1 passed, 1 total
Tests:       11 passed, 11 total
Snapshots:   0 total
Time:        6.91 s
```

Email verification

```
PASS  src/Student/Home/Home.test.js
Home Component
  ✓ renders the loading screen initially (199 ms)
  ✓ displays error message if user data cannot be fetched (28 ms)
  ✓ calls logout function when logout button is clicked (92 ms)

Test Suites: 1 passed, 1 total
Tests:       3 passed, 3 total
Snapshots:   0 total
Time:        7.101 s
```

Home page

```
  ✓ should allow removing a sibling row (3 ms)
Navigation Buttons
  ✓ renders Back and Next buttons on step 2 (2 ms)
  ✓ renders only the Submit button on step 3 (3 ms)
  ✓ renders only the Next button on step 1 (3 ms)
  ✓ calls prevStep when Back button is clicked (3 ms)
  ✓ calls nextStep when Next button is clicked (3 ms)
EducationalAttainment Component
  ✓ renders all sections and fields correctly (39 ms)
  ✓ allows user interaction with input fields and dropdowns (16 ms)
  ✓ renders a Submit button (10 ms)
AccountSettings Component
  ✓ renders all fields and button correctly (4 ms)
  ✓ allows user interaction with email and password inputs (4 ms)
  ✓ handles Change Password button click (2 ms)

Test Suites: 1 passed, 1 total
Tests:       24 passed, 24 total
Snapshots:   0 total
Time:        6.745 s
```

```
PASS src/Student/Profile/Profile.test.js
PersonalDetails Component
✓ should render all form fields correctly (262 ms)
✓ should allow input in form fields (27 ms)
✓ should submit the form when the button is clicked (26 ms)
Family Background - Step 1: Parent Information
✓ should render all form fields correctly (147 ms)
✓ should allow input in Parent 1 form fields (34 ms)
✓ should allow selection in Parent 1 education dropdown (22 ms)
✓ should allow input in Parent 2 form fields (24 ms)
Family Background - Step 2: Guardian Information
✓ should render all form fields correctly (21 ms)
✓ should allow input in the form fields (22 ms)
✓ should allow selection in the education dropdown (10 ms)
Family Background - Step 3: Siblings Information
✓ should render siblings table with correct headers (7 ms)
✓ should allow adding a sibling row (10 ms)
✓ should allow removing a sibling row (9 ms)
```

Profile pages

```
PASS src/Student/Register/Register.test.js
RegisterComponent
✓ renders first step correctly (251 ms)
✓ renders second step correctly when not verified (12 ms)
✓ renders success message when verified (7 ms)

Test Suites: 1 passed, 1 total
Tests:       3 passed, 3 total
Snapshots:   0 total
Time:        6.664 s
```

Register account

```
PASS src/Student>StatusAndScheduling/RescheduleModal.test.js
RescheduleModal Component
✓ renders correctly when open (138 ms)
✓ does not render when isOpen is false (3 ms)
✓ updates the selected date when a new date is chosen (86 ms)
✓ updates the reason textarea when text is entered (31 ms)
✓ displays an error message when present (9 ms)
✓ calls onClose when 'Cancel' button is clicked (11 ms)
✓ calls onClose when 'Confirm' button is clicked (8 ms)

Test Suites: 1 passed, 1 total
Tests:       7 passed, 7 total
Snapshots:   0 total
Time:        6.8 s
```

Reschedule enrollment date

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

```
PASS  src/Student/StatusAndScheduling/StatusAndScheduling.test.js
StatusAndScheduling Component
  ✓ renders the Header component (136 ms)
  ✓ renders the main title (18 ms)
  ✓ renders all steps in the progress tracker (23 ms)
  ✓ highlights the current step (8 ms)
  ✓ renders the enrollment date (16 ms)
  ✓ renders the rescheduling feedback (11 ms)
  ✓ opens the RescheduleModal when 'Appeal for Reschedule' is clicked (37 ms)
  ✓ closes the RescheduleModal when 'Cancel' is clicked in the modal (21 ms)
  ✓ confirms the RescheduleModal when 'Confirm' is clicked (20 ms)

Test Suites: 1 passed, 1 total
Tests:       9 passed, 9 total
Snapshots:   0 total
Time:        7.154 s
```

Status and scheduling page

```
PASS  src/Student/SubmissionAndSubject/SubmissionAndSubject.test.js
SubmissionAndSubject Component
Step 1 Tests
  ✓ renders the header and student categorization details (372 ms)
  ✓ renders the 'Next' button and changes step when clicked (40 ms)
  ✓ renders the main title (14 ms)
  ✓ renders student categorization details (27 ms)
  ✓ renders the radio buttons for student types (98 ms)
  ✓ renders the student info form correctly (24 ms)

Step 2 Tests
  ✓ renders Step 2 content after clicking 'Next' in Step 1 (23 ms)
  ✓ renders the subject table with dropdowns and inputs (62 ms)
  ✓ renders the Add Subject and Remove buttons (67 ms)
  ✓ renders the back and next buttons (18 ms)

Step 3: File Upload and Date Selection
  ✓ renders file upload sections correctly (9 ms)
  ✓ handles file uploads (5 ms)
  ✓ handles advising date selection (5 ms)
  ✓ calls handleBack when Back button is clicked (5 ms)

Test Suites: 1 passed, 1 total
Tests:       14 passed, 14 total
Snapshots:   0 total
Time:        6.528 s
```

File submission page

```
PASS  src/Student/SubmissionAndSubject/UploadRegular.test.js
Upload Regular Components
  ✓ should render the header section (137 ms)
  ✓ should render the title "Submissions and Subjects" (71 ms)
  ✓ should render the "Curriculum Checklist" section title (25 ms)
  ✓ should render two upload buttons for Curriculum Checklist and Certificate of Recognition (22 ms)
  ✓ should render feedback with name and feedback text in Curriculum Checklist (26 ms)
  ✓ should render feedback with name and feedback status in Certificate of Recognition (18 ms)
  ✓ should render "Pre-Enrollment Form" text (12 ms)
  ✓ should render and trigger "Download" button (23 ms)
  ✓ should render and trigger "Print" button (33 ms)

Test Suites: 1 passed, 1 total
Tests:       9 passed, 9 total
Snapshots:   0 total
Time:        7.589 s
```

File upload for regular students

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

```
node -e "trace-deprecation.js" --trace-warnings node ./node_modules/.bin/jest src/Student/SubmissionAndSubject/UploadIrregular.test.js
PASS  src/Student/SubmissionAndSubject/UploadIrregular.test.js
  Upload Irregular Component
    ✓ should render the header section (131 ms)
    ✓ should render the title "Submissions and Subjects" (18 ms)
    ✓ should render the "Transcript of Records" section (21 ms)
    ✓ should render the "Pre-Enrollment form section (14 ms)
    ✓ should render and trigger "Print" button (23 ms)
    ✓ should render and trigger "Download" button (13 ms)

Test Suites: 1 passed, 1 total
Tests:       6 passed, 6 total
Snapshots:  0 total
Time:        6.859 s
```

File upload for irregular students

```
PASS  src/Student/SubmissionAndSubject/SuccessRegular.test.js
  SuccessRegular Component
    ✓ renders the header section (114 ms)
    ✓ renders the main title (14 ms)
    ✓ displays the success message (8 ms)
    ✓ renders buttons with correct labels (78 ms)
    ✓ renders the animated checkmark (10 ms)

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:  0 total
Time:        7.154 s
```

```
PASS  src/Student/SubmissionAndSubject/SuccessIrregular.test.js
  SuccessIrregular Component
    ✓ renders the header section (136 ms)
    ✓ renders the main title (56 ms)
    ✓ displays the success message (12 ms)
    ✓ renders buttons with correct labels (104 ms)
    ✓ renders the animated checkmark (8 ms)

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:  0 total
Time:        6.521 s
```

Success messages

MAINTENANCE

Overview

The maintenance of the Department of Computer Studies – Cavite State University – Bacoor Campus Enrollment System is crucial to ensure its continued functionality, security, and performance. Regular maintenance activities will help address any issues that arise, implement necessary updates, and enhance the system based on user feedback. This section outlines the scope of maintenance, schedules, responsibilities, and documentation practices to facilitate system upkeep.

Maintenance Scope

The maintenance includes system updates, which will include **additional features**, **security patches and bug fixes**. **Continuous monitoring** of the system performance or any reports on the documentation logs, **user support** which provides assistance for users for any issues, and **documentation updates**.

Maintenance Schedules

To ensure the performance and reliability of the Department of Computer Studies – Cavite State University – Bacoor Campus Enrollment System, the following maintenance schedule will be implemented:

- **Before the semester:** Check if all functionalities are working, conduct system/end-user training, and provide support by instructing stakeholders on how to use the system.
- **After the semester:** Perform data archiving and cleanup to prepare the system for the next enrollment period.
- **Monthly:** Apply software updates and security patches, replace deprecated code, and review maintenance documentation logs and reports.
- **Annually:** Conduct a full system review, including analysis of user feedback.

Maintenance Responsibilities

These are the roles responsible for the maintenance of the enrollment system. Once implemented, the stakeholders or the system administrator can contact the following:

System Administrator monitors the performance and daily operations of the system. The administrator.

The **Enrollment Team** will log all the reports received from the end-users also responsible for creation of user feedback.

- Contact Information
 - Alliance of Computer Scientists (ACS)
 - Information Technology Society (ITS)
- Email: cvsu.its@gmail.com

Database Administrator handles database backups and ensures data integrity

- Contact Information
 - Matthew Isaac L. Bides
- Email: matthewbides03@gmail.com

Technical Support Team provides user support and collects feedback for improvements.

- Contact Information
 - Keith Gabriell F. Dasalla
- Email: keithdasalla@gmail.com

Development Team implements feature enhancements and updates documentation.

- Contact Information
 - Jerald V. Labalan
 - Alexander James Ian J. Fernandez
- Email: jeraldlabalan26@gmail.com
- Email: fernandezalexander1107@gmail.com

*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

- Michael Angelo I. Muega
Email: bc.michaelangelo.muega@cvsu.edu.ph
- James Andrei V. Orcullo
Email: bc.jamesandrei.orcullo@cvsu.edu.ph

Maintenance Documentation Logs and Reports

Maintenance logs and reports are given by the enrollment team (society officers, advisers and system administrator) at least once a month. The format are as follows:

Maintenance Log

Date	Activity Description	Responsible Person	Duration	Notes/Comments
DD/MM/YY	Description of maintenance activity	Name and Role of the Person	HH:MM	Any relevant notes or comments

Incident Report

Date	Reported by	Description of the Incident	Status	Follow-up Required
DD/MM/YY	Name of user	Brief description of the issue	Open/Closed	Yes/No

CONCLUSION

The Cavite State University - Bacoor City Campus Department of Computer Studies has developed a web-based application system aimed at addressing the inadequacies of crowd control and information dissemination and administrative efficiency by adding features such as enrollment and advising date designation, file uploads, notifications, PDF generation and printing, and enrollment queuing status to reduce the manual administrative tasks faced during the enrollment process.

To enhance the application, the following are recommended:

- **Online Payment Option.** Enables users to make secure payments directly through the application. This feature could support various payment methods, including credit/debit cards, online banking, and mobile wallets. It would streamline transactions for tuition fees, society fees and miscellaneous fees.
- **Integration with Student Portal.** Integrate the application with a student portal where users can access personalized information, such as course schedules, grades, attendance records, and assignment submissions.
- **Event Calendar.** Include a calendar feature that displays important enrollment-related events, deadlines, and academic calendars.
- **Activity Log on notifications.** Other than the announcements on the notifications, add an activity log to track user interactions.

APPENDICES

GLOSSARY OF TERMS

Adviser. A user role in the system, responsible for counterchecking, accepting and rejecting file submissions such as Last Semester's Certificate of Recognition, Transcript of Records, and Curriculum Checklist.

Agile Scrum Methodology/Agile Model. Iterative approach to software development that emphasizes collaboration and adaptability. It is the type of Software Development Life Cycle (SDLC) Model that is used in the development of the system.

Certificate of Registration. An official document issued by the campus registrar that certifies the legitimacy of a student as a bona fide member of the campus.

Data Form. Describes the collection of personal information from students. Typically, the Department of Computer Studies (DCS) organizations use this form for event attendance purposes and for student record management by the Office of Student Affairs and Services (OSAS).

Department of Computer Studies. The department responsible for overseeing the two technology-related programs at Cavite State University - Bacoor City Campus: Information Technology and Computer Science.

Enrollee. A user role in the system that refers to students who have not yet enrolled and are currently undergoing the enrollment process.

Incoming First Year Student. Categorized in the system as student type 6 (S6), are the incoming first year students that passed the university's admission and examinations.

Irregular Student. Categorized in the system as student type 3 (S3), are students who are not taking the recommended academic load due to failed requirements or employment commitments.

Pre-Enrollment Form. A form that includes all the subjects the student wishes to enroll in for the semester. It is provided once the student has completed the advising stage.

Regular Student. Categorized in the system as student type 1 (S1), are students that are taking the normal academic load.

Returnee Student. Categorized in the system as student type 5 (S5), are students who have previously enrolled in a program at the institution but have taken a break or leave of absence and are now returning to continue.

Shiftee Student. Categorized in the system as student type 6 (S6), are students that changed their academic program to another.

Society Officer. A user role in the system that is in charge of setting the enrollees' date of enrollment, updates the enrollment status of each enrollee, and issues receipts once the enrollee is paid on the society membership fee.

Society Fee Receipt. Issued by a Society Officer, it includes the enrollee's name and the amount paid, which is generated in the system.

Student. Refers to the enrollees once they completed the entire enrollment process.

System Administrator. A user role in the system that is responsible for issuing Certificate of Registration (COR) and adding other users in the enrollment team.

Transferee Student. Categorized in the system as student type 6 (S6), are students who have transferred from another Cavite State University (CvSU) campus, taking the same program. If a student does not meet this criteria, they will be considered an irregular student (S3).

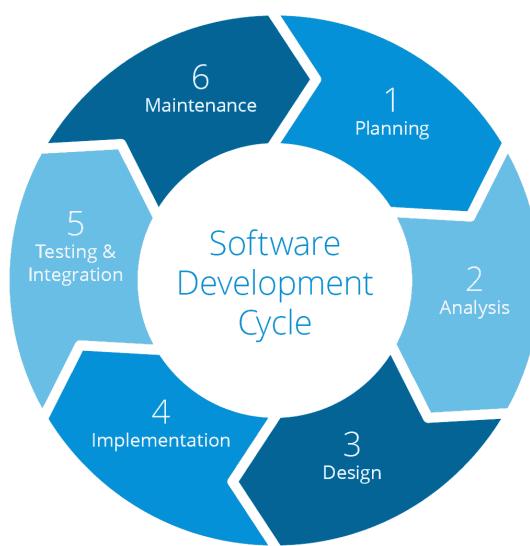
REFERENCES

Inflectra. (n.d.). What are use cases? Retrieved January 14, 2025, from <https://www.inflectra.com/Ideas/Topic/Use-Cases.aspx>

Lucidchart. (n.d.). Agile software development life cycle. Lucidchart. Retrieved January 14, 2025, from <https://www.lucidchart.com/blog/agile-software-development-life-cycle>

ADDITIONAL DIAGRAMS AND TABLES

AGILE SDLC USED (Lucidchart, n.d.)



*Enrollment System for Department of Computer Studies –
Cavite State University – Bacoor Campus*

GANTT CHART

