SECJ2203: Software Engineering

System Documentation (SD)

UTM Ethics Approval Management System (EAMS)

Version 1.0

Date: 25 / 11 / 2024

Faculty of Computing

Prepared by: Huperrush

Revision Page

a. Overview:

This document represents **Version 1.0** of the system documentation, comprising two main sections: **Introduction** and **Specific Requirements**. The Introduction section outlines the project's purpose, scope, definitions, acronyms and abbreviations as well as references to relevant sources. The Specific Requirements section provides detailed information about user characteristics, system features, use case details, performance and other requirements as well as design constraints. Together, these sections establish a clear foundation for understanding the system and guiding its development.

b. Target Audience:

UTM staff, student and staff UTM REC.

c. Project Team Members:

Member Name	Role	Task	Status
ABDALLA ALI ABDALLA ALI A23CS3022	Moderator	Monitor group progress	Complete
OTHMAN HASSAN OTHMAN ALI A23CS3026	Recorder	Record the results of discussion	Complete
EYAD AIMEN ELSHEIKH KHALIL A23CS3024	Skeptic	Give query and comment on the assignment	Complete
CHEW CHUAN KAI A23CS0062	Accuracy checker & Reporter	Check and compile the assignment	Complete

d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
1.0	ABDALLA ALI ABDALLA ALI	Completed Chapter 1 and 2, Section Introduction and Specific Requirements	25/11/2024

Table of Contents

1. Introduction:

1.1 Purpose:

This System Documentation (SD) serves to outline the requirements and features of the proposed UTM Ethics Approval Management System (UTM EAMS). It focuses on the System Requirements Specification (SRS), which defines what the system will do and how it is expected to perform. The goal of this documentation is to transition the traditional manual processes of UTM REC into an efficient and automated solution that enhances the overall effectiveness, accuracy, and usability of the research management process.

By leveraging detailed use cases and sequence diagrams, this SD provides a clear understanding of system workflows, interactions, and requirements. These tools help identify potential issues and areas for improvement during development, ensuring alignment with stakeholder needs. The SRS also highlights both functional and non-functional requirements, addressing key aspects such as system performance, usability, and scalability.

This documentation is designed to meet the needs of its primary stakeholders, including **UTM students**, **UTM staff**, and **UTM REC staff**, by capturing their expectations and ensuring the system delivers a user-centric, reliable, and high-performing research management solution. It provides a structured foundation for monitoring progress and ensuring the system meets all requirements efficiently.

1.2 Scope:

The software product is UTM Ethics Approval Management System (UTM EAMS) which is designed to streamline and automate the research ethics approval process for UTM REC. This system caters to three key stakeholders: **UTM students**, **UTM staff**, and **UTM REC staff**, ensuring a seamless and efficient experience for managing research applications. The system offers several essential features, including secure user registration, application submission, payment processing, real-time tracking, and communication modules.

Existing users can log in with their UTM credentials, while new users are required to register using their UTM email, matric number, or staff ID. Verification is enforced via email, ensuring only authorized members gain access. Applicants must also upload identification ocuments and agree to UTM REC's ethical guidelines, emphasizing the importance of

compliance with research policies. Once registered, users are assigned roles with specific access permissions based on their identities.

The system allows applicants to submit research applications via an online portal, where fields are auto-validated to minimize errors. Applications are categorized by research type (e.g., clinical, non-clinical, or animal research) and routed to the appropriate subcommittees for review. Secure payment functionality is integrated, enabling applicants to pay fees through third-party gateways such as e-wallets, online banking, or debit cards. Payment statuses are updated instantly, and receipts are emailed to applicants while maintaining a comprehensive transaction log.

A dashboard provides applicants with real-time updates, including application progress, estimated processing times, and pending actions. Automated email notifications keep users informed at each stage. For high-risk applications, the system notifies applicants to schedule pitching sessions, which are managed collaboratively between applicants and staff.

For UTM REC staff, the system automates routine tasks such as email communication and payment verification, freeing time for more critical responsibilities. Staff can track application progress, facilitate high-risk reviews, and provide live chat support for applicants. The system ensures that all updates are synchronized across dashboards for both applicants and staff, improving transparency and efficiency.

In addition to these features, the system incorporates a live chat and chatbot module to assist users with common inquiries or escalate complex issues to staff. Our goal is to develop a research management system that enhances operational efficiency, ensures data accuracy, and elevates user satisfaction. By transitioning from manual processes to an automated solution, this system aligns with UTM REC's mission to uphold research integrity and streamline the ethics approval process.

1.3 Definitions, Acronyms and Abbreviation:

Term	Definition
UTM EAMS	The software product being developed to manage the research integrity and streamline the ethics approval process.
SRS	System Requirements Specification - a document that outlines the requirements and objectives of the software product being developed.
SD	System Documentation
UTM REC	UTM Research Ethics Committee
ID	Identity Document

1.4 References:

[1] Nishadha. (2015, February 17). *Use Case Diagram Relationships Explained with Examples*. Creately Blog. https://creately.com/blog/diagrams/use-case-diagram-relationships/

[2] Visual Paradigm. (2019). *What is Sequence Diagram?* Visual-Paradigm.com. https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/

[3] Visual Paradigm. (2019). *What is Activity Diagram?* Visual-Paradigm.com. https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/

[4] *Domain Modeling: What you need to know before coding*. (2021, September 29). Thoughtworks. https://www.thoughtworks.com/insights/blog/agile-project-management/domain-modeling-what-you-need-to-know-before-coding

[5] Altexsoft. (2023, March 28). *Technical Documentation in Software Development: Types and T*. AltexSoft. https://www.altexsoft.com/blog/technical-documentation-in-software-development-types-best-practices-and-tools/

[6] altexsoft. (2022, July 26). *Non-functional Requirements: Examples, Types, How to Approach*. AltexSoft. https://www.altexsoft.com/blog/non-functional-requirements/

1.5 Overview:

This System Documentation (SD) outlines the specific requirements gathered from stakeholders for the development of our proposed software. It covers key aspects such as user characteristics, system features, use case details, performance and other requirements as well as design constraints.

For user characteristics, we detail the intended users of the system, including their technical expertise, knowledge levels, and physical abilities. The system features are explained through various diagrams, such as use case diagrams, activity diagrams, sequence diagrams, domain model class diagrams and state machine diagrams, providing a comprehensive understanding of the system's functionality.

Performance and other requirements as well as design constraints are elaborated in detail within this document. The SD is structured to follow the Software Development Life Cycle (SDLC) phases, including requirement analysis, design and testing, with each phase thoroughly documented using diagrams and descriptive explanations.

2. Specific Requirements:

2.1 User characteristics:

In UTM EAMS we have 2 main users of the system, which are: applicants can be a UTM student or UTM staff member and staff REC.

2.1.1 Applicant:

- The applicants must at least be doing a master degree.
- The applicants who will use the software are expected to have basic computer skills, including familiarity with web-based applications.
- The applicants are expected to have experience in email communications.

2.1.2 Staff REC:

- The staff REC are expected to have experience with web-based applications.
- Some of the staff REC are expected to need software training on how to use the new system.
- The staff REC are expected to know how to provide help for the applicants.

2.2 System Features

The UTM EAMS is a web-based application software, this system is accessed via web and internet. The system will allow the applicants to apply, submit and track their research in one platform. The system will also provide one platform for staff REC to process, update all the research applications and notify the applicant if there are any changes required.

The system features are illustrated in Figure 2.1 below. The detailed description of each module and functions is tabulated in Table 2.1.

Figure 2.1: Use Case Diagram for UTM EAMS

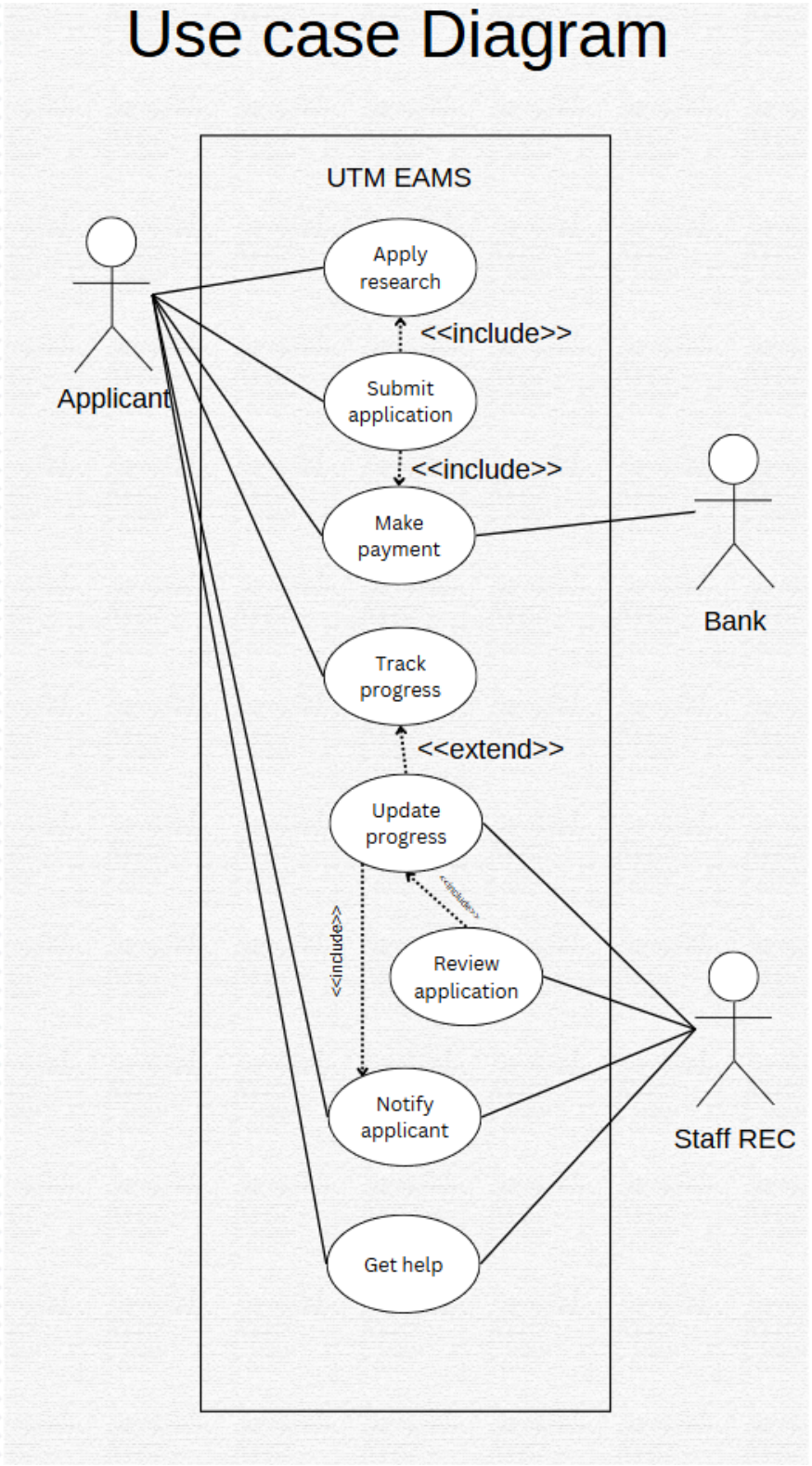

Table 2.1: Description of Module and Functions for UTM EAMS

Module	*Function*	*Description*
	UC001 - Apply research	This use case allows the applicants to fill in the research application form
	UC002 - Submit application	This use case allows the applicants to submit the research application form.
	UC003 - Make payment	This use case allows the applicants to pay for the research application form via third party gateway
	UC004 - Track progress	This use case allows the applicants to track their submitted research applications and see all the updates regarding the reviewing process.
	UC005 - Update progress	This use case allows the staff REC to update the status of the research in the reviewing process.
	UC006 - Review application	This use case allows the staff REC to view all the research information and review it.
	UC007 - Notify applicant	This use case allows the staff REC to notify the applicant of the updates regarding the applicant research.
	UC008 - Get help	This use case allows the applicants to get help regarding their issues in the system, they will be assisted either with the system chatbot or one of staff REC members.

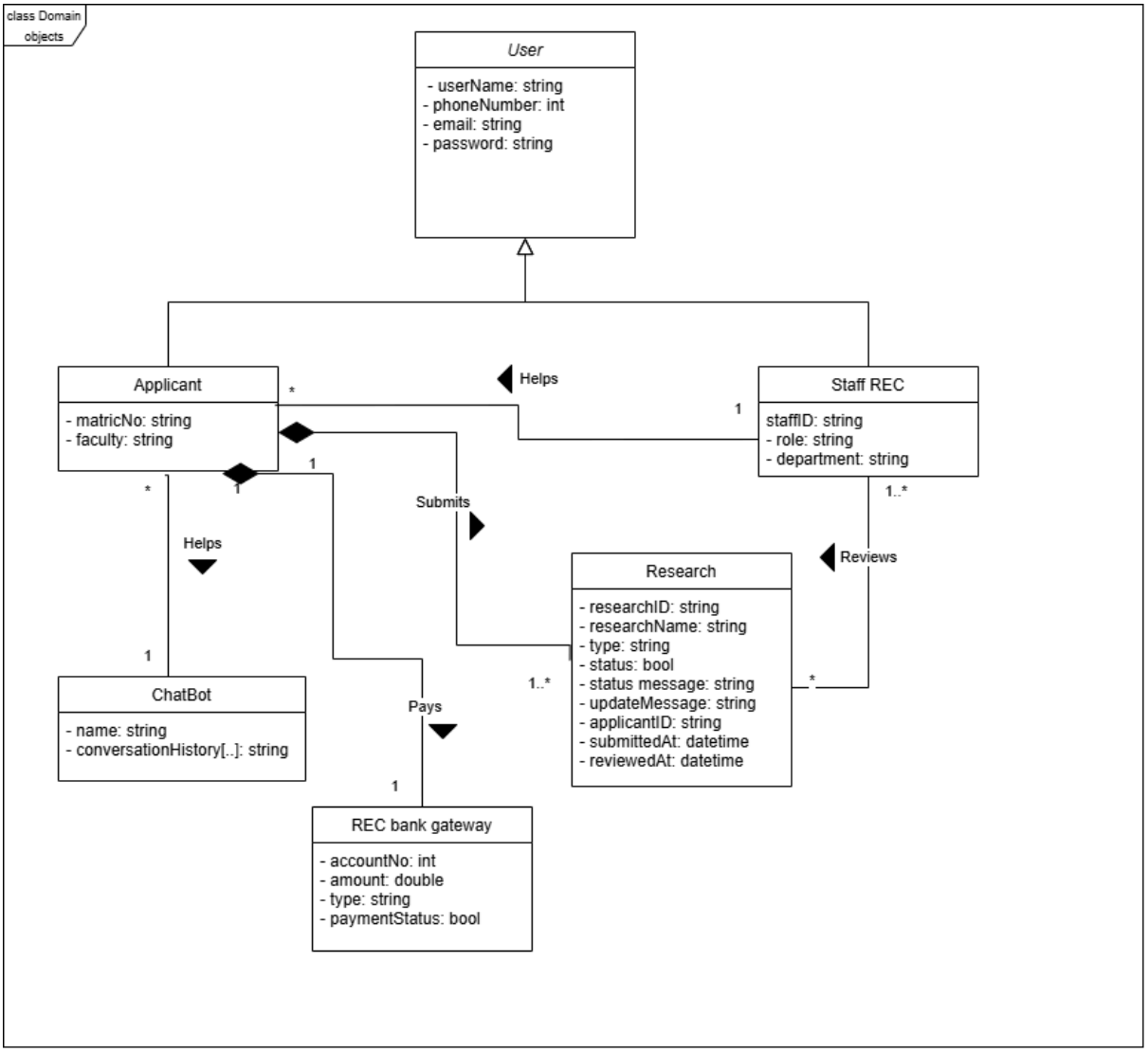

Figure 2.2: Domain Model for UTM EAMS

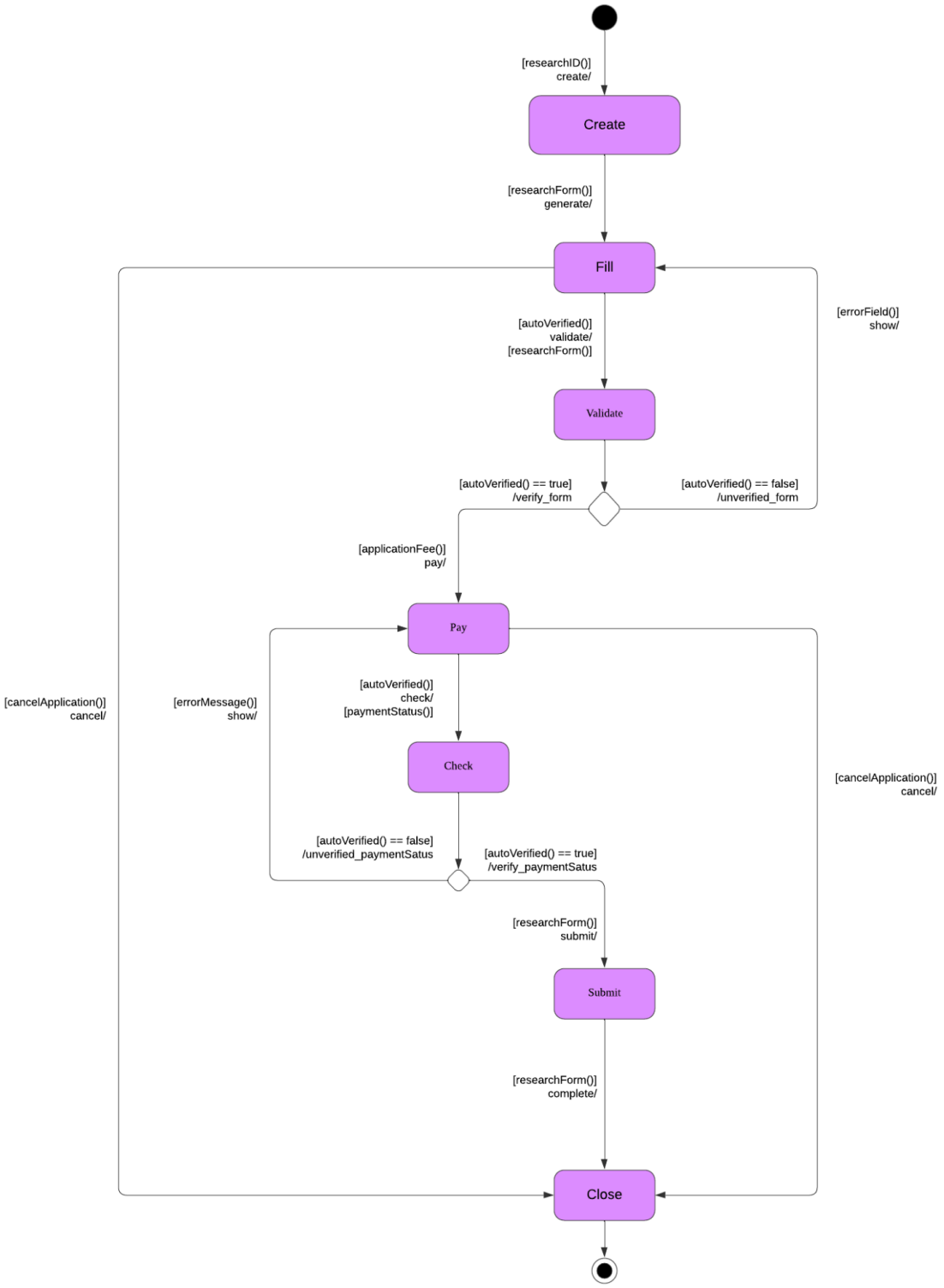

Figure 2.3: State Diagram for Research class

2.3 Use Case Details:

2.3.1 UC001: Use Case Apply research:

Table 2.3.1: Use Case Description for Apply research

Use case: Apply research
ID: UC001
Actors: Applicant (consists of Student and UTM Staff)
Includes: UC002 Submit application
Preconditions: Applicant is already registered in the system
Flow of events: 1. The system provides the applicant a sample form to help the applicant fill in the form 2. The applicant fills in the required information. 3. The system checks the input fields. 4. if there was an error in input fields 4.1 for each input field that has an error 4.1.1 The system displays an error message 4.1.2 The applicant fills in the required information 5. Include (Submit application)
Alternative flow: The applicant cancels the application.
Postconditions: The system returns the applicant to the home page.

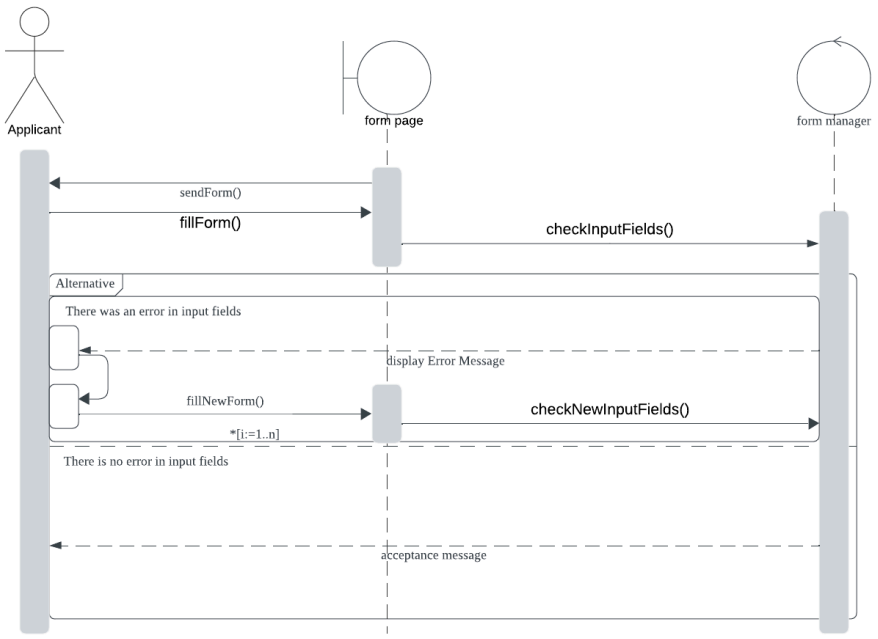

Figure 2.3.1: Sequence Diagram for Apply research

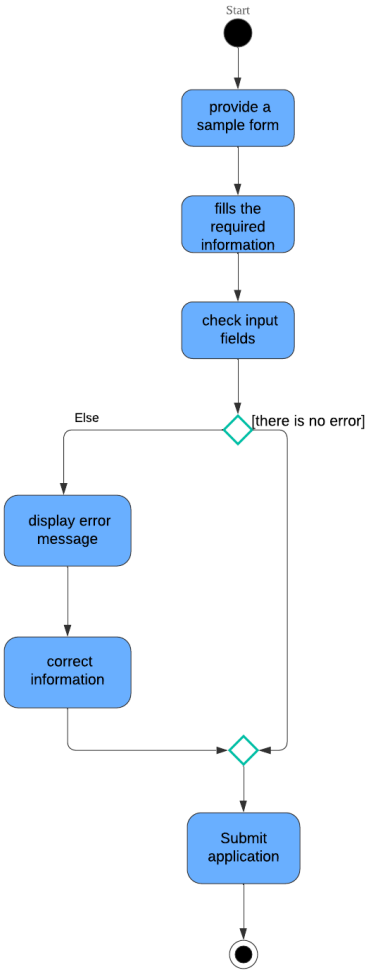

Figure 2.3.2: Activity Diagram for Apply research

2.3.2 UC002: Use Case Submit application

Table 2.3.2: Use Case Description for Submit application

Use case: Submit application
ID: UC002
Actors: Applicant (consists of Student and UTM Staff)
Includes: UC003 Make payment
Preconditions: The application form fields are error free.
Flow of events: 1. The applicant clicks the "Submit" button. 2. Include (Make payment). 3. The system categorizes the application based on research clinical, non-clinical and animal research. 4. The application is routed to the respective subcommittee for review. 5. The applicant receives an email notification confirming successful submission.
Postconditions: The applicant's dashboard is updated to reflect the submission status and next steps.

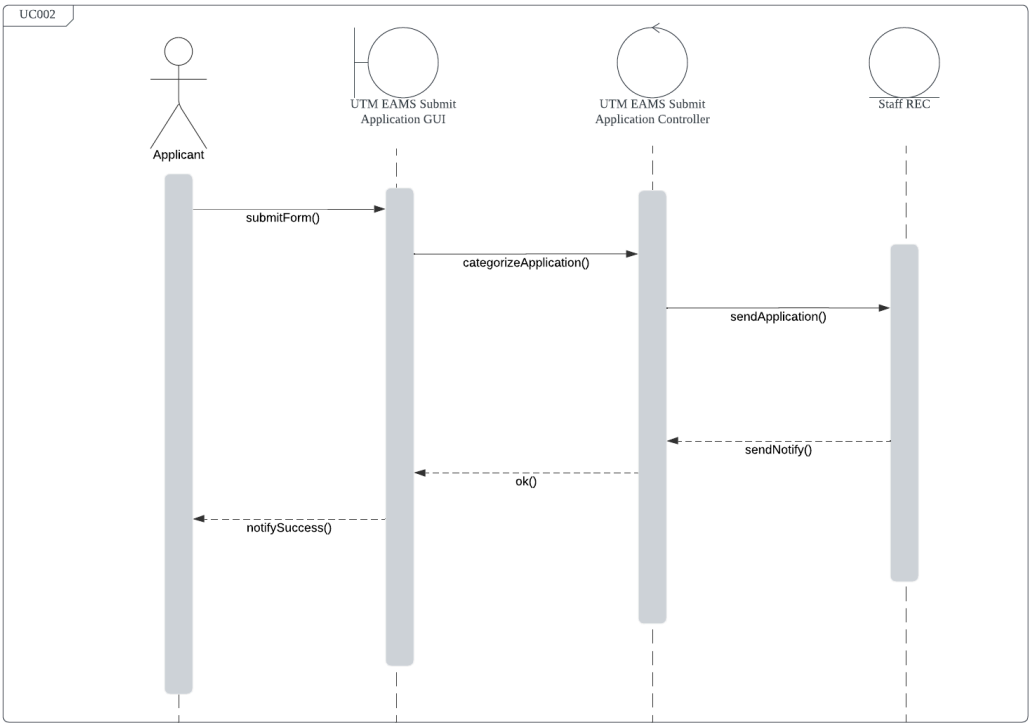

Figure 2.3.3: Sequence Diagram for Submit application

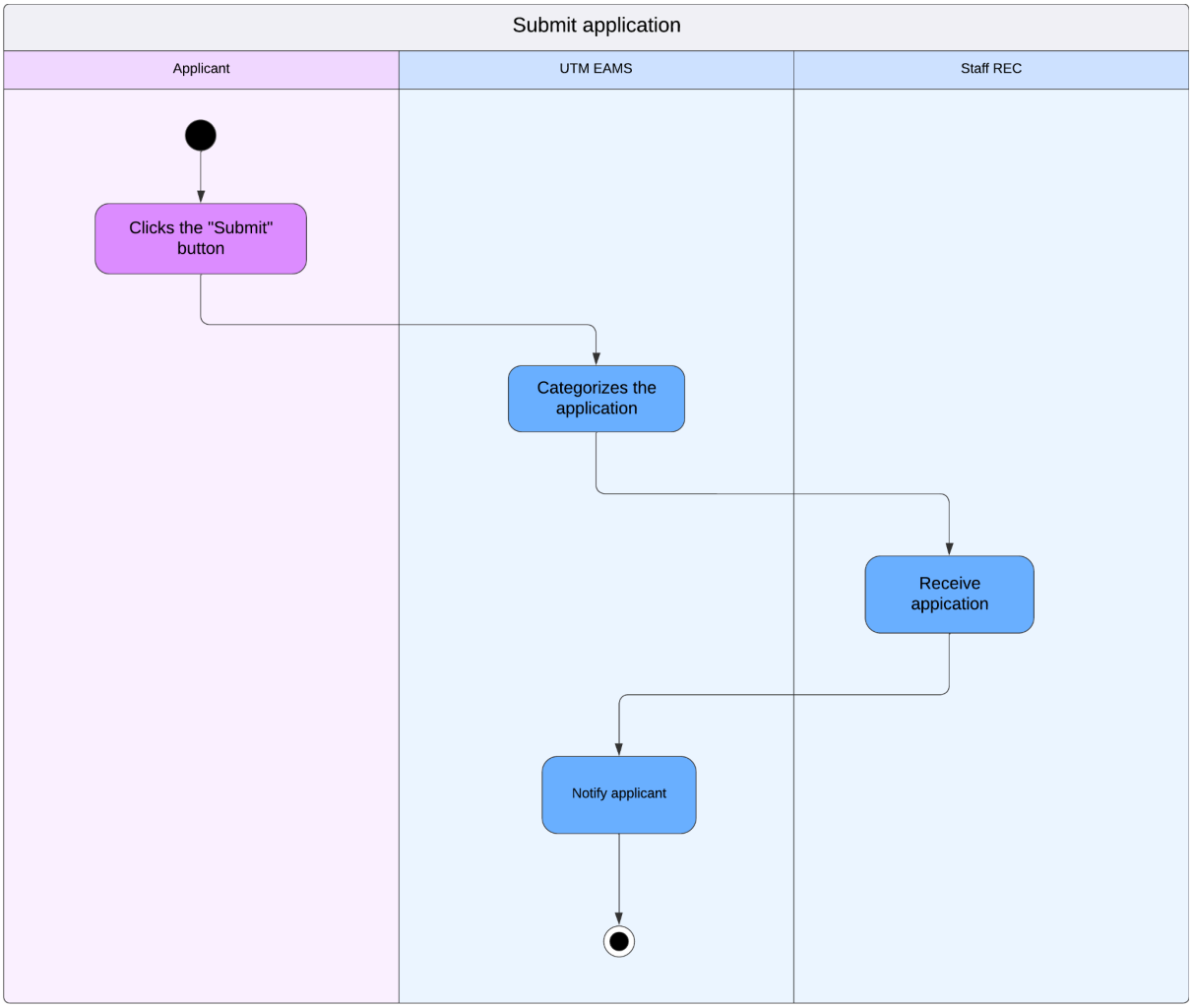

Figure 2.3.4: Activity Diagram for Submit application

2.3.3 UC003: Use Case Make payment

Table 2.3.3: Use Case Description for Make payment

Use case: Make payment
ID: UC003
Actors: Applicant (consists of Student and UTM Staff) Bank
Preconditions: The application form has been completed and validated.
Flow of events: 1. The system redirects the applicant to a third-party payment gateway. 2. The applicant chooses a payment method either e-wallet, online banking or debit card. 3. The applicant completes the transaction. 4. If there is a technical issue with the payment gateway. 4.1. The system advises the applicant to try again later. 5. If the transaction is fail 5.1. The system alerts the applicant with an error message “Transaction Fails”. 6. If the transaction is successful 6.1. The payment gateway processes the payment and sends a confirmation to the system. 6.2. The system updates the applicant’s record with the payment status. 6.3. An email receipt is sent to the applicant for confirmation.
Postconditions: 1. Payments status is updated as “Completed” in the system. 2. The applicant’s dashboard reflects the updated payment status.
Alternative flow: At any given time the applicant has the ability to cancel the research application

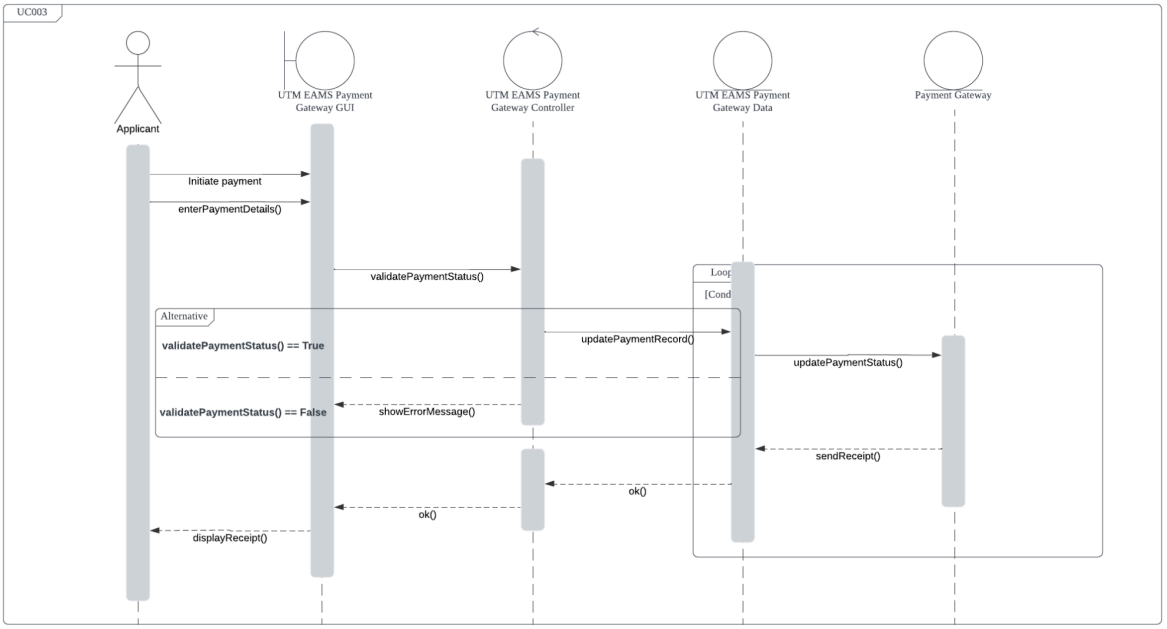

Figure 2.3.5: Sequence Diagram for Make payment

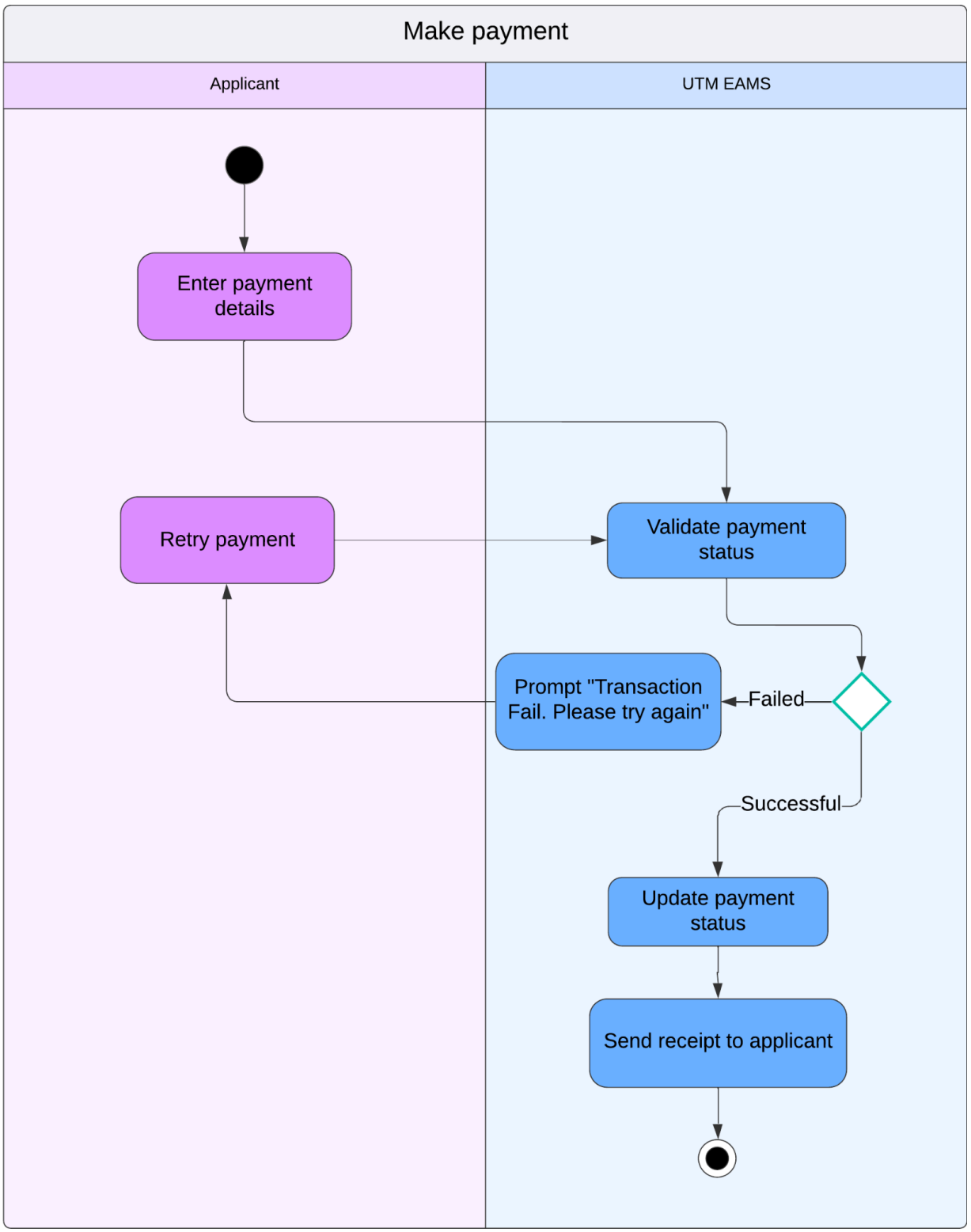

Figure 2.3.6: Activity Diagram for Make payment

2.3.4 UC004: Use Case Track progress:

Table 2.3.4: Use Case Description for Track progress

Use case: Track progress
ID: UC004
Actors: Applicant (consists of Student and UTM Staff)
Extension points: <Updated progress>
Preconditions: 1. The applicant must submit the research.
Flow of events: 1. The applicant opens the dashboard. 2. The applicant selects the research that wants to track. 3. The system checks if there were any updates from the UTM REC staff on the research. 4. If <Updated progress>, then 4.1 The system tells the applicant that there was an update and displays an update message. 5. else 5.1 The system displays there are no updates message 6. The system displays estimated processing time and previous updates on the research
Postconditions:

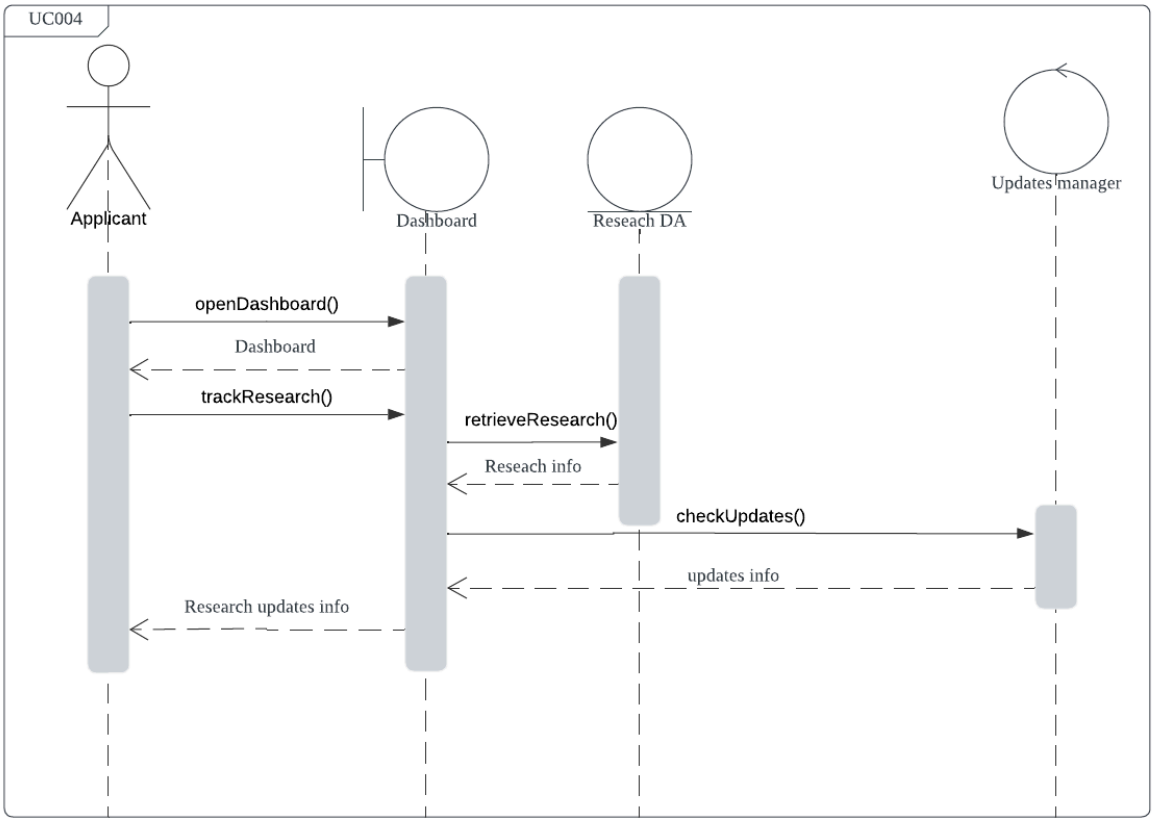

Figure 2.3.7: Sequence Diagram for Track progress

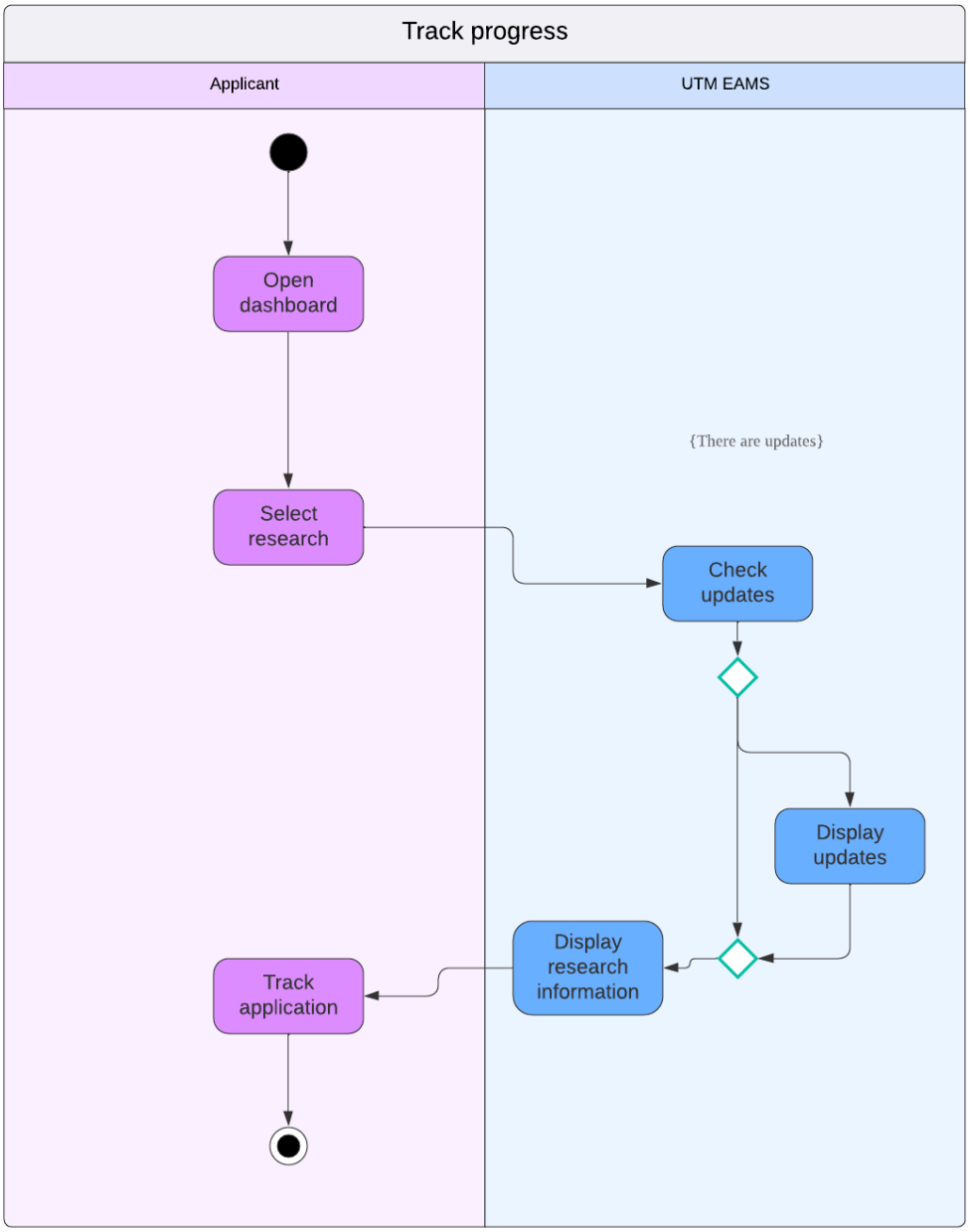

Figure 2.3.8: Activity Diagram for Track progress

2.3.5 UC005: Use Case Update progress:

Table 2.3.5: Use Case Description for Update progress

Use case: Update progress
ID: UC005
Actors: Staff REC
Includes: UC007 Notify applicant
Extends: UC004 Track progress at <Updated progress>
Preconditions: 1. The staff REC must review the submitted research
Flow of events: 1. The Staff REC selects the status flag. 2. The staff REC writes the status update message. 3. Include(Notify applicant). 4. The system displays a message indicating that the update is a success.
Alternative flow: The system fails to submit or save the updated status.
Postconditions: The staff REC has to resubmit the updated progress.

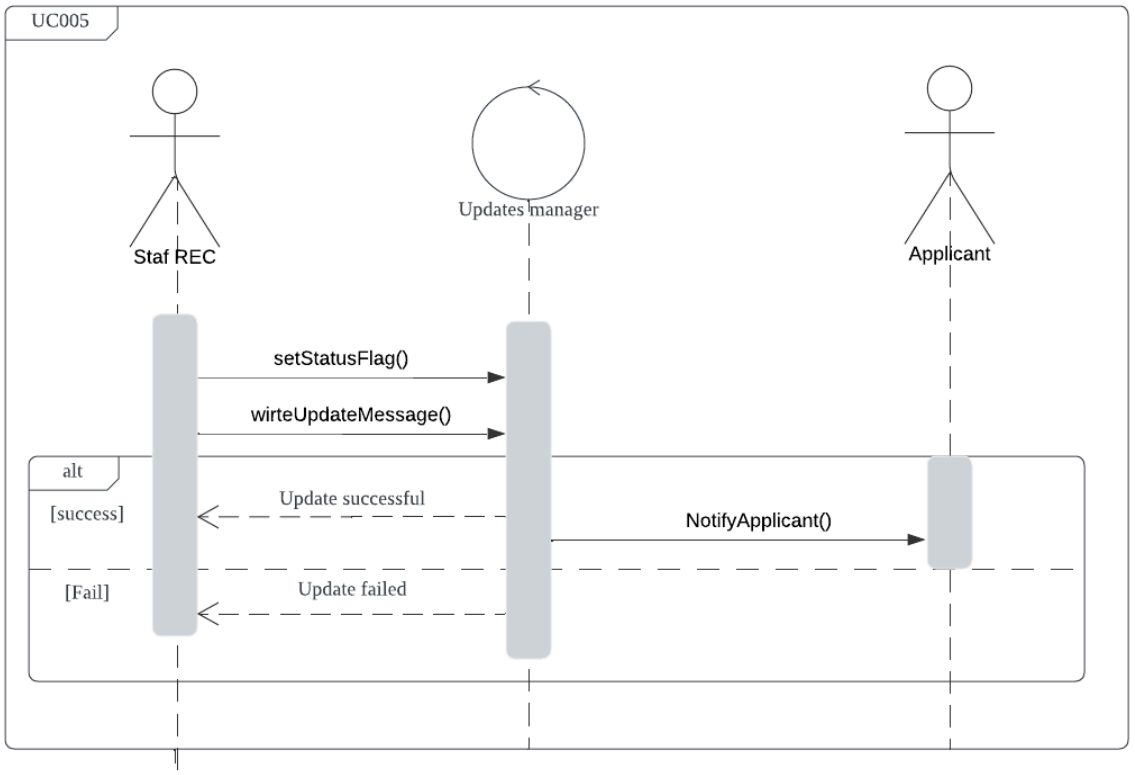

Figure 2.3.9: Sequence Diagram for Update progress

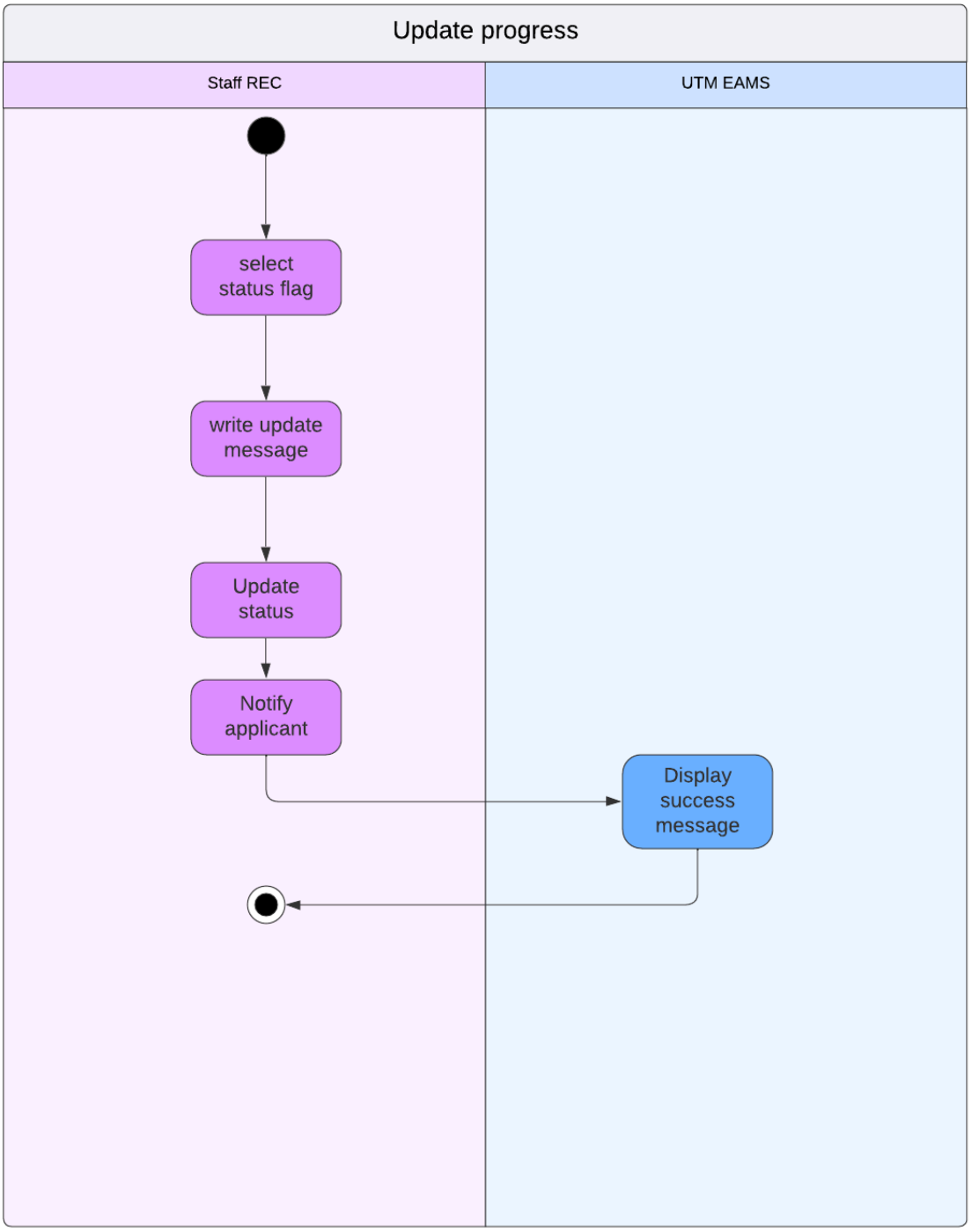

Figure 2.3.10: Activity Diagram for Update progress

2.3.6 UC006: Use Case Review application:

Table 2.3.6: Use Case Description for Review application

Use case: Review application
ID: UC006
Actors: Staff REC
Includes: UC004 Update progress
Preconditions:
Flow of events: 1. Staff REC reviews the application. 2. if Staff REC detects high-risk research applications. 2.1 The system automatically sends email to applicants to select the available time slot for the review session. 3. Include(Update progress).
Preconditions: 1. The updates should be reflected in the applicant dashboard.

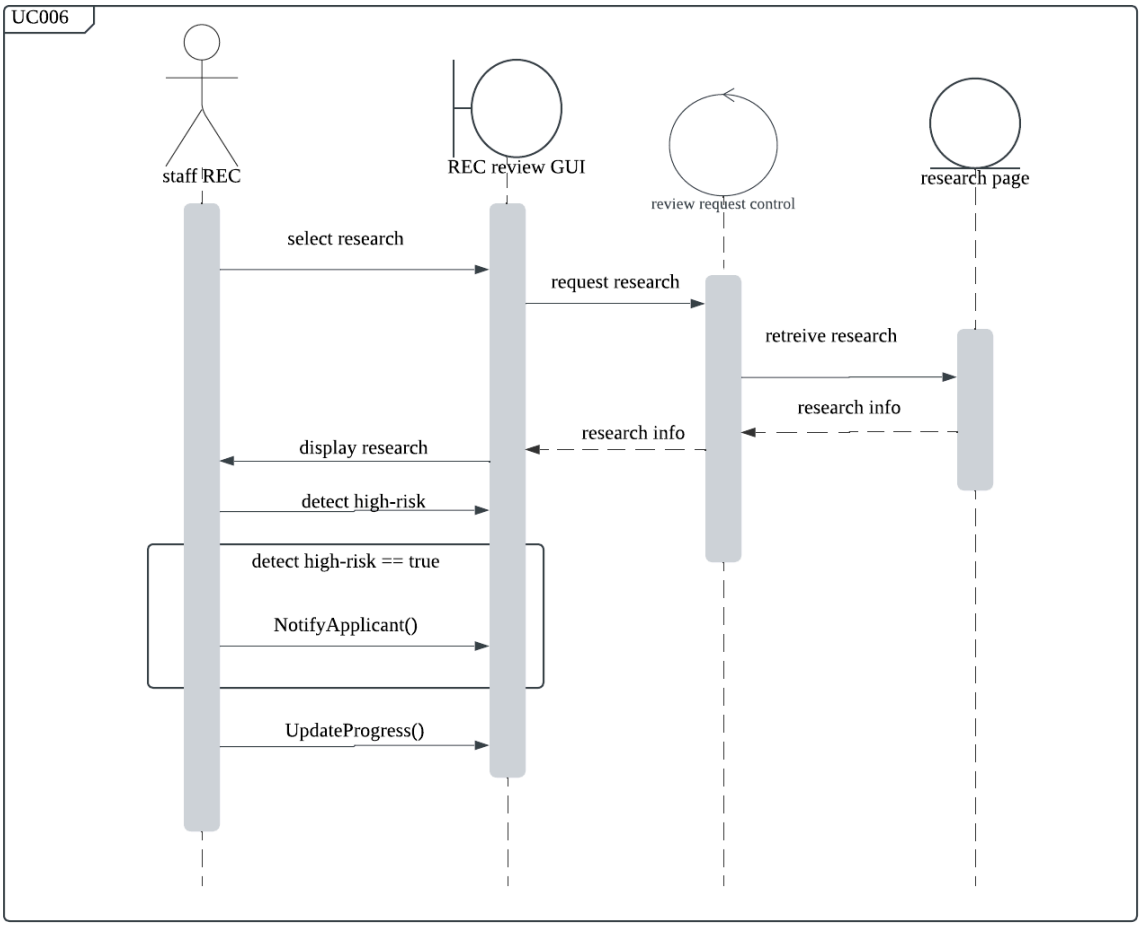

Figure 2.3.11: Sequence Diagram for Review application

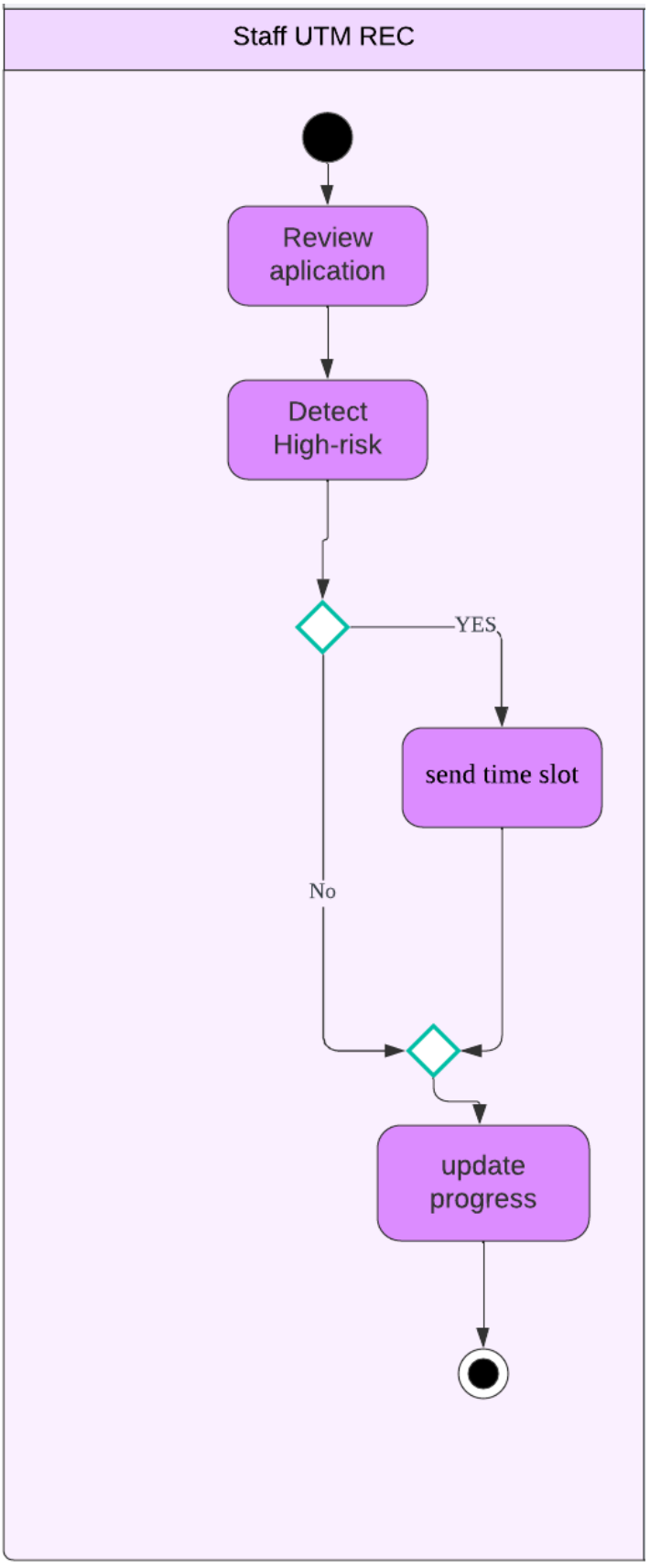

Figure 2.3.12: Activity Diagram for Review application

2.3.7 UC007: Use Case Notify applicants:

Table 2.3.7: Use Case Description for Notify applicants

Use case: Notify applicants
ID: UC007
Actors: Staff REC Applicant
Preconditions: 1. The system progress is updated
Flow of events: 1. staff REC sends notification to the applicant's email regarding the updated progress. 2. applicants receive the email notification indicating that the progress is updated.
Postconditions:

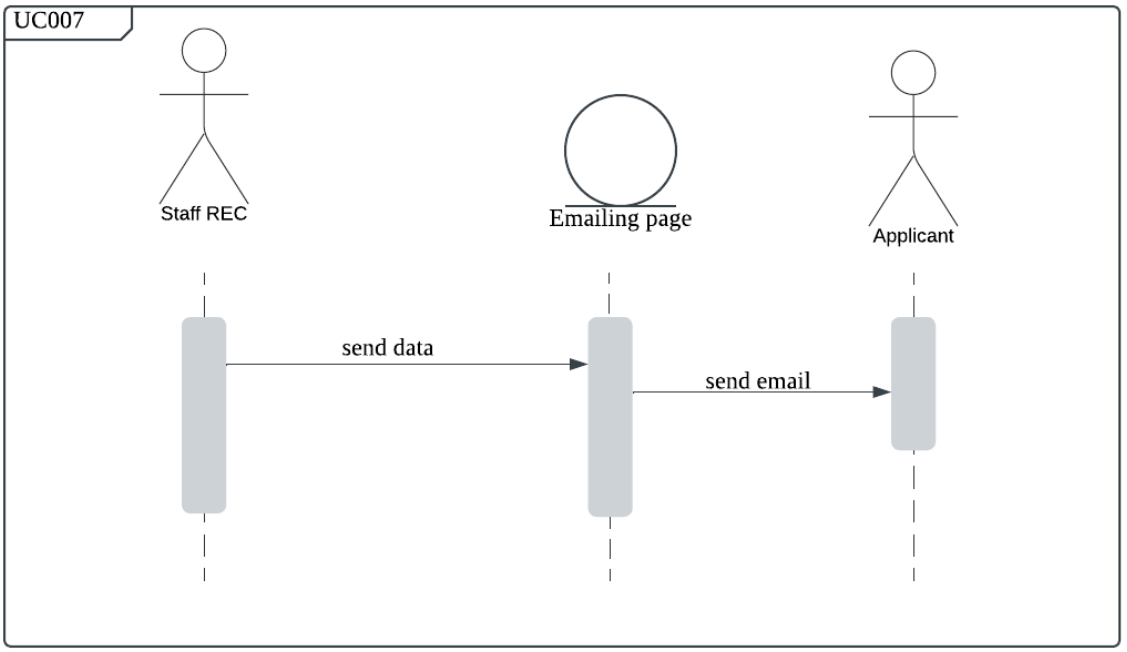

Figure 2.3.13: Sequence Diagram for Notify applicants

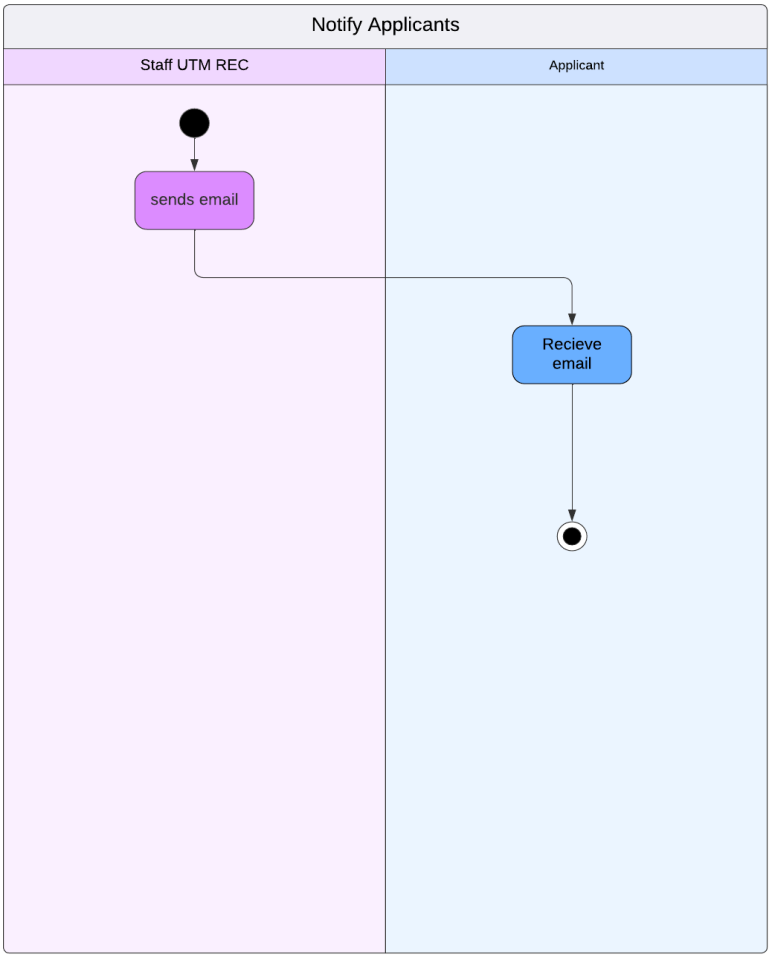

Figure 2.3.14: Activity Diagram for Notify applicants

2.3.8 UC008: Use Case Get help:

Table 2.8: Use Case Description for get help

Use case: Get help
ID: UC008
Actors: Staff REC Applicant
Preconditions:
Flow of events: 1. The applicant faced a problem and asked for help. 2. The system will try to address the applicant problem via chatbot. 3. If the chatbot could not solve the applicant problem, then 3.1 The system will forward the applicant problem to the appropriate staff REC 3.2 The staff REC will contact the applicant and solve the applicant's problem
Postconditions:
Alternative flow: 1. The staff REC could not help the applicant
Postconditions: 1. The staff REC arranges a meeting with the applicant.

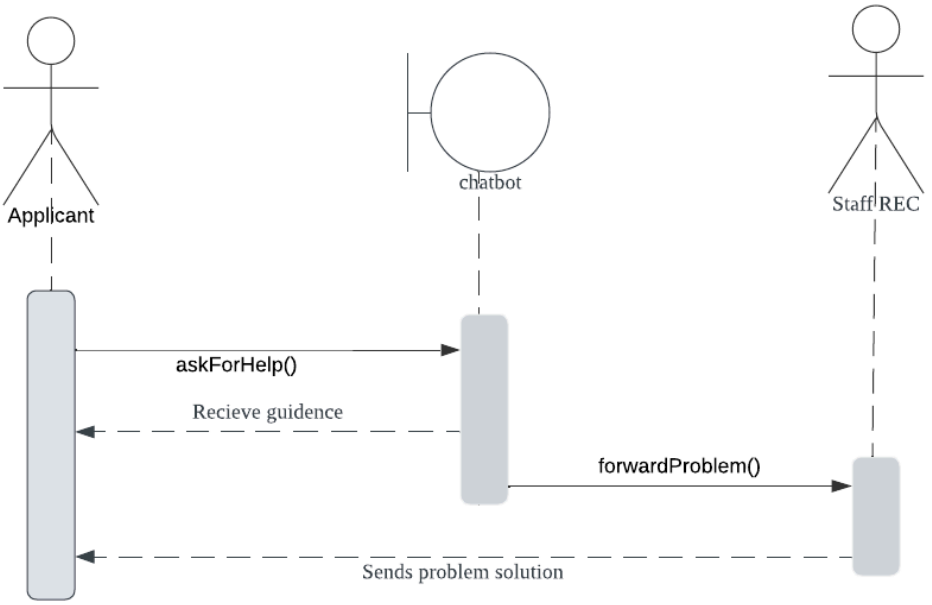

Figure 2.3.15: Sequence Diagram for Get help

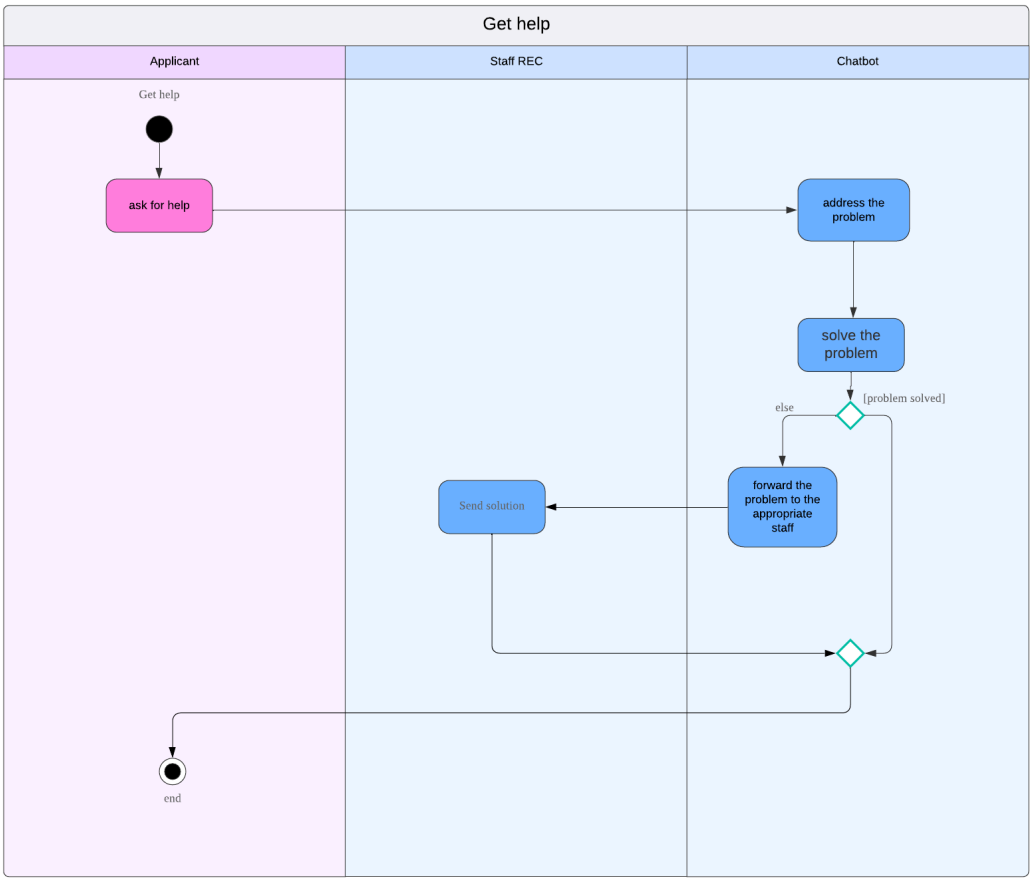

Figure 2.3.16: Activity Diagram for Get help

2.4 Performance and Other Requirements:

Software System Attributes:

- Usability: The system must be designed user-friendly to provide ease of use, featuring a clear and clean interface for all users.
- Reliability: The system must perform its functions accurately and consistently, minimizing errors or bugs.
- Maintainability: The system must be designed to be easily modified, repaired, and enhanced to accommodate future requirements or updates.
- Compatibility: The system must function seamlessly with Windows operating systems (Windows 10 and above) Android devices (version 10 and above) iOS devices (version 13 and above).

Performance:

- Response Time: The system should respond to user requests within a maximum of 5 seconds.
- Capacity: The system must support up to 500 concurrent user requests.

Other Requirements:

- Security: The system and its data must be protected from unauthorized access and malicious attacks to ensure confidentiality, integrity, and availability.
- Legal and Regulatory: The system must comply with all relevant laws, regulations, and standards applicable in Malaysia.

2.5 Design Constraints:

Below are some of the constraints that the UTM EAMS must adhere:

- **Technology stack constraints:** The system uses specific programming languages, frameworks and technologies such as Java, .NET, MySQL as mandated by the organization.
- **Deployment constraints:** The system must be deployable on UTM's existing infrastructure without requiring significant upgrades or additional servers.
- **Data retention constraints:** The system must store and retain user data for a predefined period (example: 8 years) before automatic deletion, as per institutional policies.
- **Localization constraints:** The system must support the Malay and English languages to cater to diverse users within Malaysia.
- **User environment constraints:** The system must operate efficiently in environments with limited internet bandwidth or intermittent connectivity.