Bilkent University
Department of Computer Engineering
CS-319 Final Deliverables
12 November 2024

Bilkent Üniversitesi

Fall 2024-2025

Rida Fatima
Emine Fidan
Mert Özkaya
Fazlı Güdül
Ömer Yaslıtaş

Table of Contents

Non-functional Requirements

1.1 Usability

The **UI/UX design** is crucial because it ensures that the system is easy to use for all types of users, including high school counselors, students, guides, and coordinators. A good design will help users complete their tasks without confusion or frustration. When users can quickly find what they need, it improves their experience and makes the system more efficient.

- Following Nielsen's Heuristics Guideline, use simple and clean design elements to avoid clutter.
- Color-code important statuses like accepted (green), pending (yellow), and rejected (red) for clarity.
- Use a minimum font size of 16 px for readability.
- Ensure that buttons for actions, like "Submit" or "Confirm," are large and easy to spot.
- Use high contrast between background and text to make reading easier.
- Make action buttons stand out with bright colors like red or orange.
- Add simple visual effects like slight shadows on buttons to guide users on where to click.

1.2 Security

Our system will handle sensitive user data, such as information about counselors, students, and guides, so keeping this data secure is crucial to ensure privacy and compliance with legal standards. We must protect it from unauthorized access, prevent data loss, and safeguard against web attacks.

- Passwords will be encrypted before storing in the database.
- Only authorized users will be allowed to access sensitive information.
- Daily backups will be implemented to prevent data loss in case of system failure.
- Deleted user data will be fully removed from the system to protect privacy.
- Session data will be cleared upon logout, requiring re-authentication to access secure areas.

1.3 Performance

Our system must operate efficiently to allow users, such as counselors, students, and guides, to complete tasks swiftly and without delays, especially during peak usage times.

- The system should handle up to **500 concurrent users** smoothly, even during high-traffic periods like the start of the academic year.
- **Logins** should be completed within **5 seconds** when the correct credentials are provided.
- Actions such as **submitting a tour application** or **retrieving available tours** must be processed in under **3 seconds**.
- **Tour status updates** must reflect across the system in real time, with a delay of no more than **2 seconds**.

1.4. Maintainability

Our system needs to be easily maintainable to ensure that updates, bug fixes, and feature additions can be implemented quickly without disrupting the user experience. This will help keep the system running efficiently and minimize downtime.

- We will use **modular code structure** to ensure that different components (e.g., user roles, tour management) are independent, making updates and fixes more manageable.
- Each part of the system will follow a **standardized documentation format,** with sections for purpose, input/output, dependencies, and examples.
- **Bug fixes and updates** should be applied without taking the system offline for extended periods, allowing the rest of the system to remain operational during maintenance.
- **Version control** will be used to track changes and ensure smooth rollbacks if an issue arises with new updates.

1.5. Scalability

Our system must be scalable to handle increasing numbers of users and data as Bilkent's tours and activities grow. This will ensure that the system can continue to perform efficiently as demand increases over time.

- The system should be able to **scale horizontally** to support **up to 500 concurrent users** initially, with flexibility to scale further if demand grows beyond expectations, ensuring smooth performance during peak periods.
- The initial **database capacity** will be set to handle up to **10,000 records** for tours, users, and logs. This will ensure that the system has enough capacity for the current data load with a buffer for growth.

- The system should be prepared to scale and store **data growth** at an approximate rate of **10% annually**. This includes new user records, tour logs, and other operational data generated by the system.
- The system's **infrastructure** should support future enhancements, like adding new features or modules, without requiring major overhauls or affecting current performance.
- **Load balancing** should be implemented to distribute the workload evenly across servers, preventing bottlenecks during peak usage times.

1.6 State Diagram Integration

Integrating state diagrams ensures that the system behaves consistently in various user scenarios, which improves overall reliability and user satisfaction.

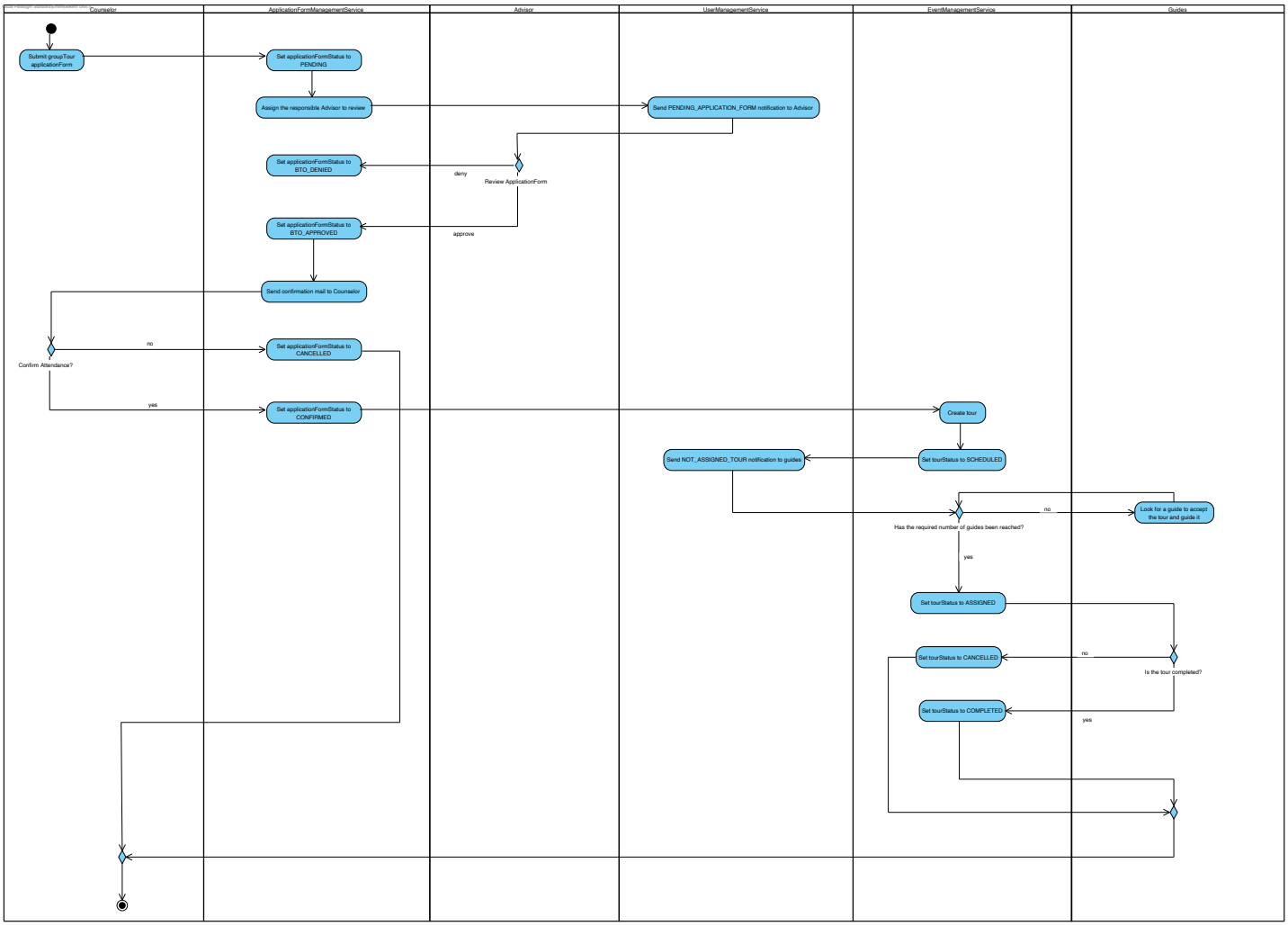
1.6.1 Application Form Submission:

- Idle: The system should immediately load the form page when accessed.
- Filling Form: As users enter data, each entry should be saved without delay to prevent data loss.
- Submitting Form: Form submissions should be processed and saved securely, ensuring no data loss.
- Pending Approval: The system should regularly update the user on the status of their application.
- Approved/Rejected: Once a decision is made, the user should be promptly notified.

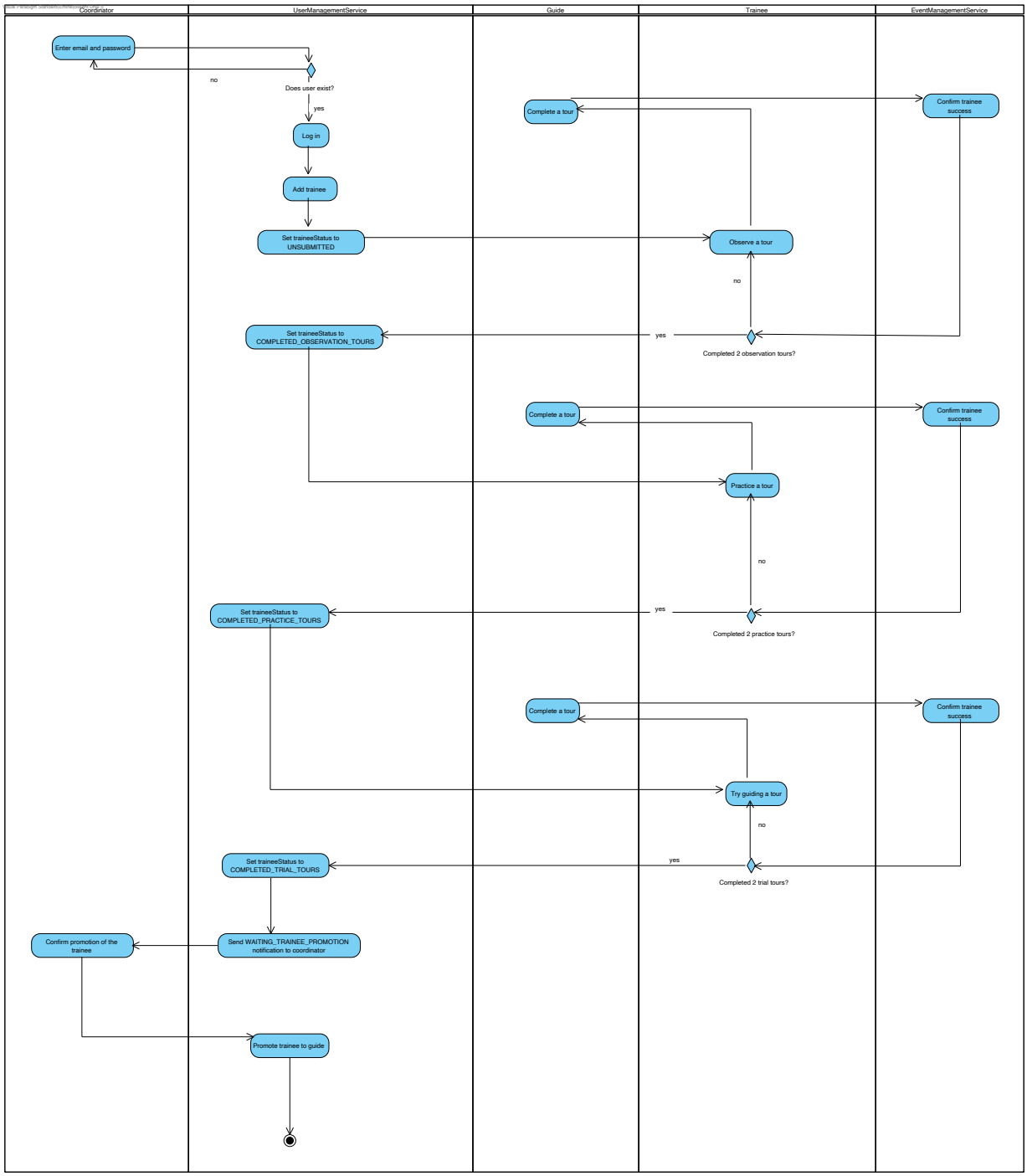
1.6.2 Guide Assignment Process:

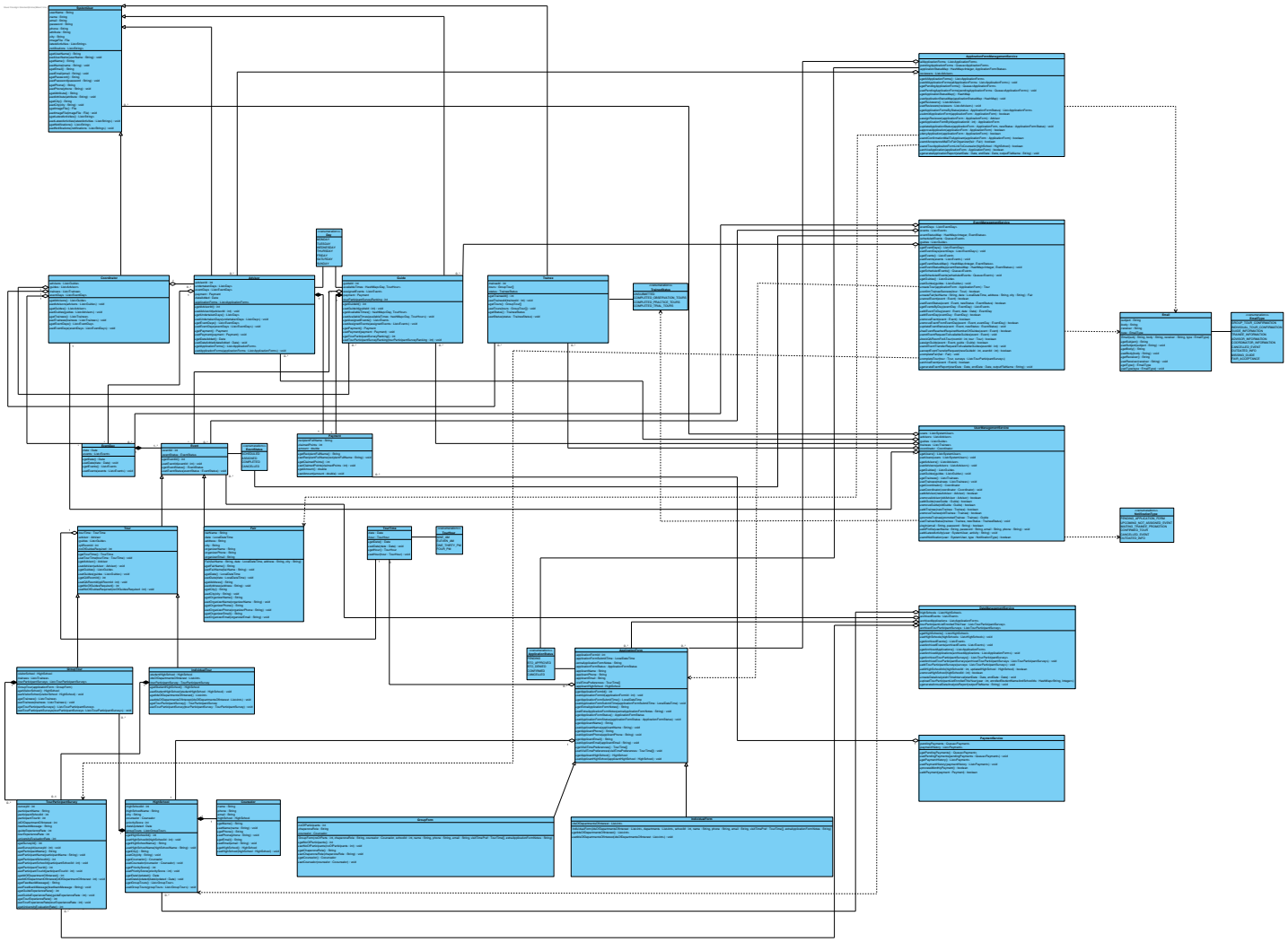
- Tour Created: Tours should enter the queue for guide assignment upon creation.
- Waiting for Guide: The system should notify guides about unassigned tours if no immediate assignment occurs.
- Guide Assigned: Upon guide assignment, the system should update tour status and notify all relevant parties.
- Tour Completed: After completion, the tour should be marked as finished, with status updates for all involved.

Activity Diagram 1
Group Tour Application and Execution Process

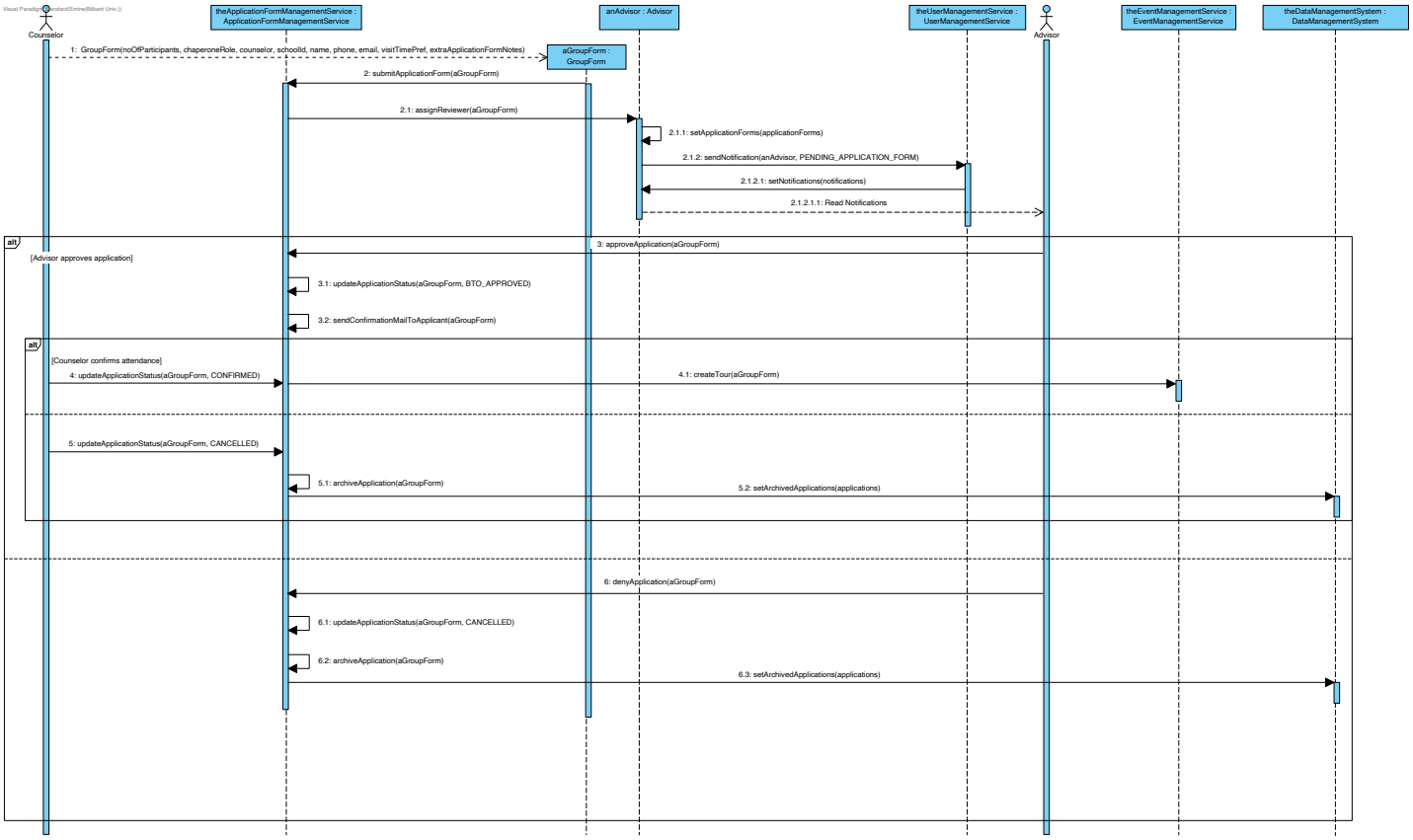


Activity Diagram 2
Trainee Promotion Process

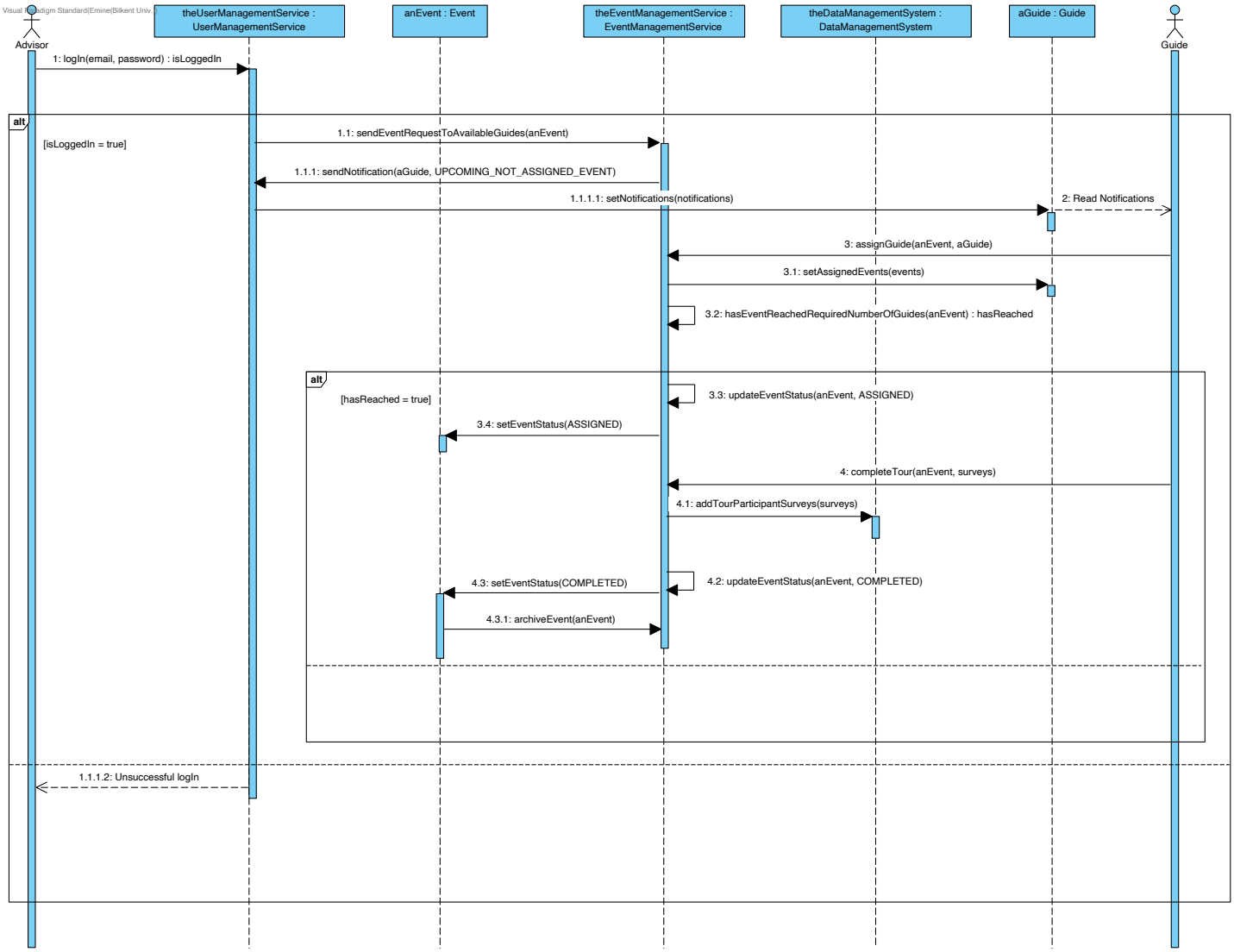




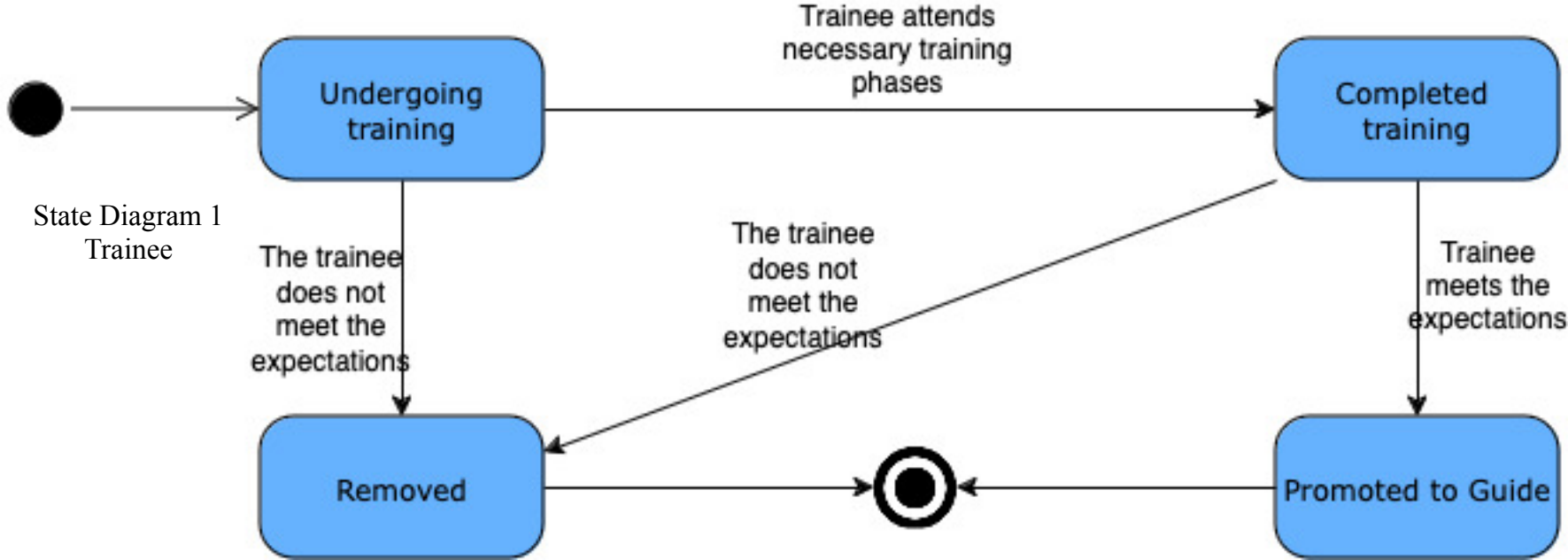
Class Diagram



Sequence Diagram 1
Application Form (GroupForm) Review Process

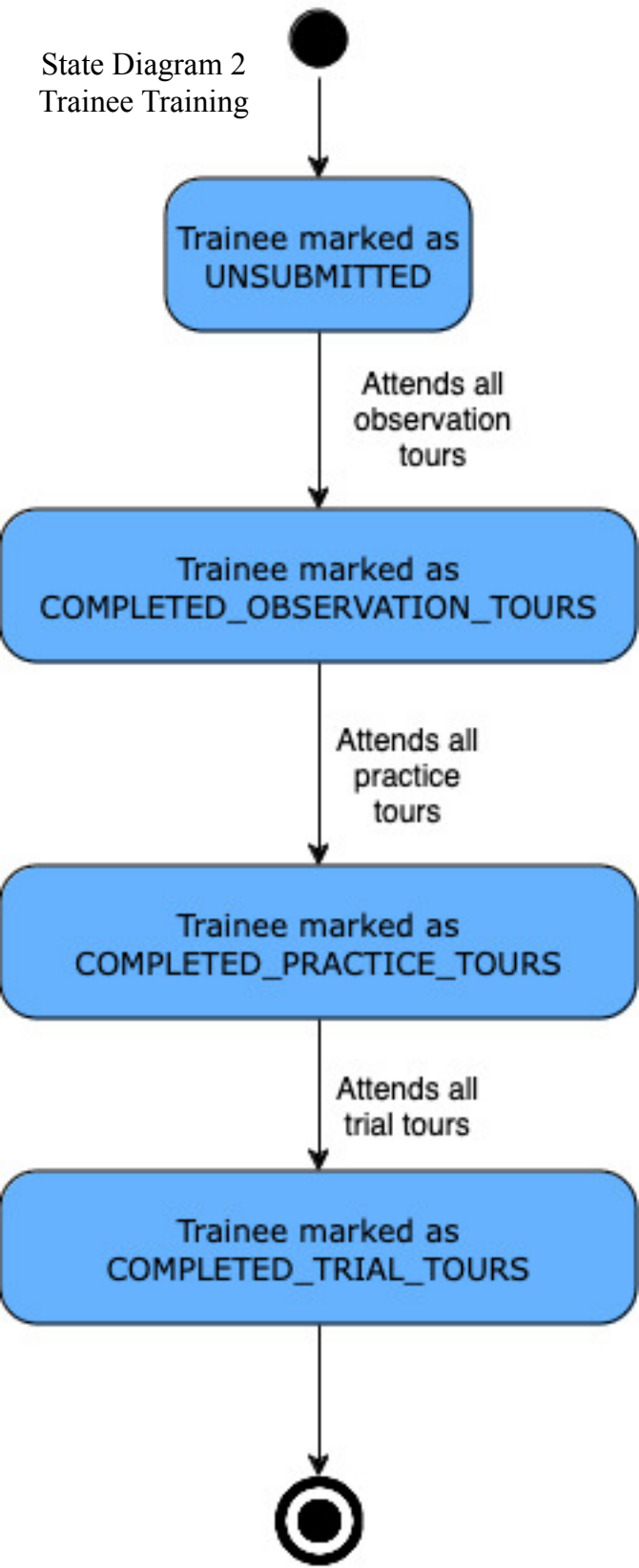


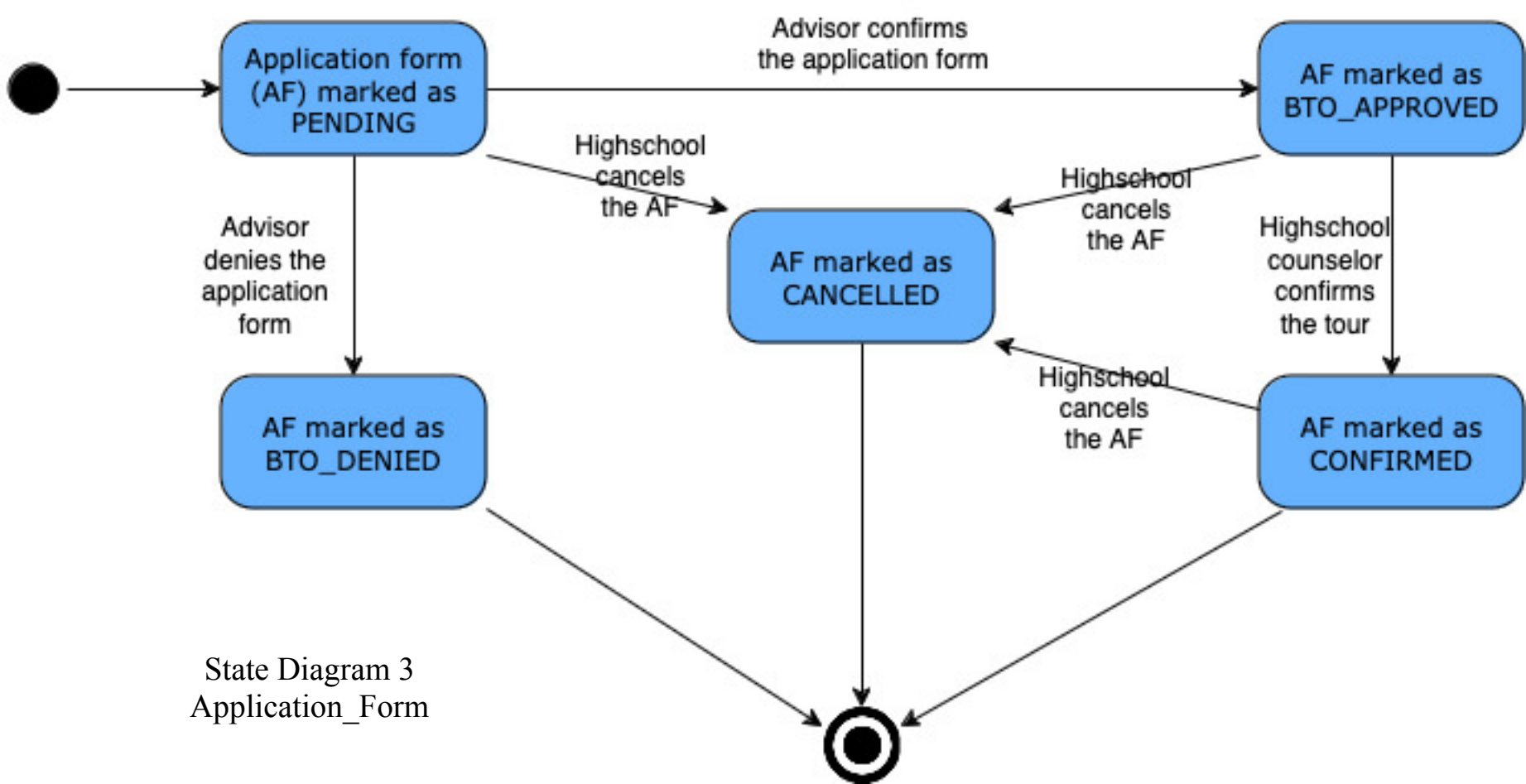
Sequence Diagram 2
Event Completion Process



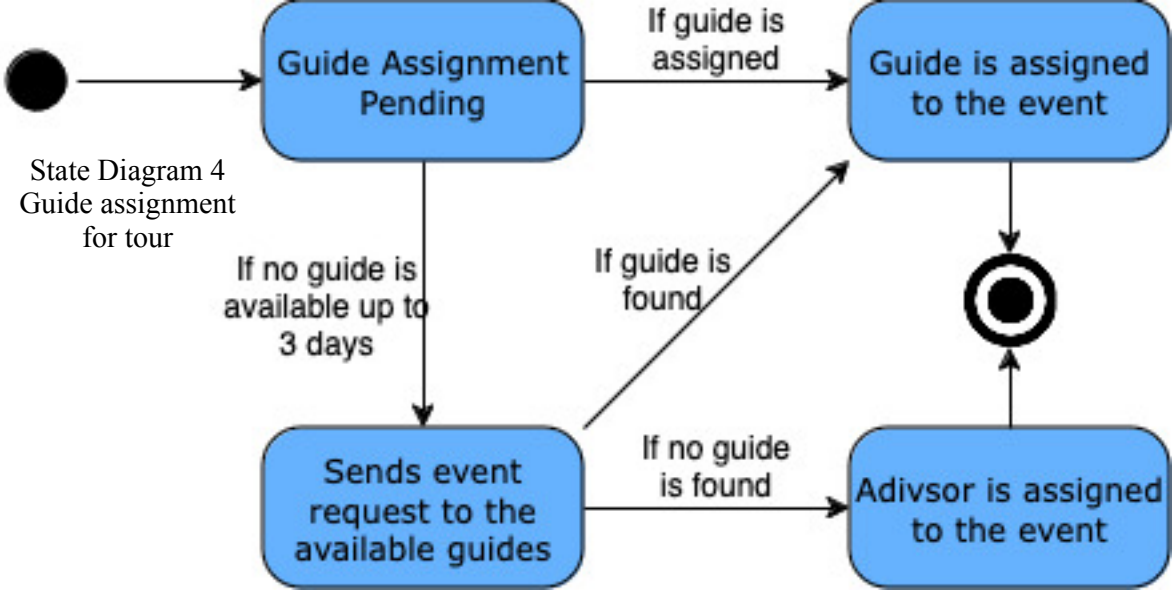
State Diagram 1
Trainee

State Diagram 2
Trainee Training

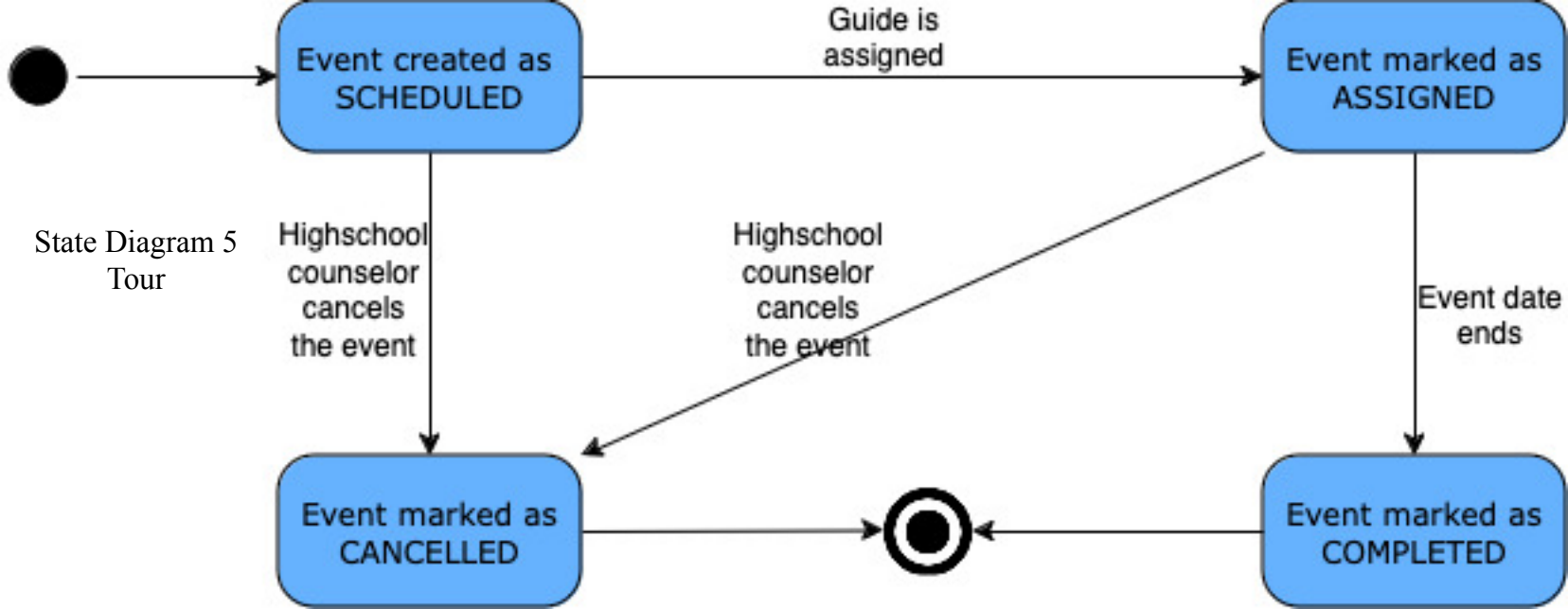




State Diagram 3
Application_Form



State Diagram 4
Guide assignment for tour



State Diagram 5
Tour

Mockups

Table of Contents: