

H-AdminSim: A Multi-Agent Simulator for Realistic Hospital Administrative Workflows with FHIR Integration

Jun-Min Lee

KAIST, Republic of Korea

LJM565@KAIST.AC.KR

Meong Hi Son

Samsung Medical Center, Republic of Korea

MEONGHI.SON@SAMSUNG.COM

Edward Choi

KAIST, Republic of Korea

EDWARDCHOI@KAIST.AC.KR

Abstract

Hospital administration departments handle a wide range of operational tasks and, in large hospitals, process over 10,000 requests per day, driving growing interest in LLM-based automation. However, prior work has focused primarily on patient–physician interactions or isolated administrative subtasks, failing to capture the complexity of real administrative workflows. To address this gap, we propose H-ADMINSIM, a comprehensive simulation framework that combines realistic data generation with multi-agent-based simulation of hospital administrative workflows. These tasks are quantitatively evaluated using detailed rubrics, enabling systematic comparison of LLMs. Through FHIR integration, H-ADMINSIM provides a unified and interoperable environment for testing administrative workflows across heterogeneous hospital settings, serving as a standardized testbed for assessing the feasibility and performance of LLM-driven administrative automation.

Data and Code Availability All data used in this study were synthetically generated. The code is publicly available on our project page.¹

Institutional Review Board (IRB) This research does not require IRB approval.

1. Introduction

Hospital administration serves as the patient’s first point of contact and handles a wide range of administrative tasks, including medication, insurance, and diagnostic inquiries, as well as department assignment and outpatient scheduling (Litchfield et al., 2017;

Bhati et al., 2023). Larger hospitals often process more than 10,000 patient requests per day (Lin et al., 2020; Wan et al., 2024). Although larger hospitals employ more administrative staff, prior studies have reported that several administrative staff members lack formal medical training (Oh et al., 2019; Lin et al., 2025). Unlike returning patients, individuals visiting a hospital for the first time with a given health condition often lack prior medical referrals, making accurate department assignment challenging even for experienced administrative staff (Kajiwara et al., 2017) and potentially leading to delays in appropriate care. Moreover, appointment scheduling requires administrative staff to coordinate physician availability and patient preferences within the hospital system while handling ongoing requests from patients with existing appointments, further increasing administrative complexity (Gupta and Denton, 2008).

Given these operational complexities, large language model (LLM)-based agents offer a promising avenue for automating or assisting hospital administrative workflows. However, their safe deployment requires prior validation in simulation environments that closely replicate real-world processes. Despite this need, existing healthcare simulation research has largely focused on patient–physician interactions, either through dialogue-based role-play for clinical education and training (Wang et al., 2024; Li et al., 2024b) or simulated clinical dialogues for evaluating LLM behavior (Fan et al., 2025; Almansoori et al., 2025). Even studies incorporating administrative components have primarily addressed physician-facing scenarios (Yan et al., 2024) or narrow subtasks, often overlooking time-dependent scheduling workflows that are central to hospital administration (Bao et al., 2024). Furthermore, although patient and physician characteristics

1. https://ljm565.github.io/h_adminsim_project/

differ markedly across hospital levels (Physiopedia, 2022; Keiser University, 2024), few studies have comprehensively modeled these differences.

To address these gaps, we propose H-ADMINSIM, a framework for realistic dialogue-based simulation of outpatient hospital administrative workflows with three core components: (1) simulation environments for hospitals of varying scales, from primary to tertiary care; (2) dialogue-based patient intake simulations that capture diverse patient backgrounds, modeled using 194 disease–symptom pairs and information on whether the patient has prior referrals; and (3) dialogue-based scheduling simulations for new appointments, rescheduling, and cancellation that reflect individual scheduling preferences. Additionally, our simulation environment follows the Fast Healthcare Interoperability Resources (FHIR) standard (Bender and Sartipi, 2013), enabling interoperability across heterogeneous institutions. While previous FHIR-based research has largely focused on API-based benchmarking (Jiang et al., 2025) or data transformation (Ben-nett et al., 2023; Carbonaro et al., 2025), to the best of our knowledge, H-ADMINSIM is the first to leverage FHIR for dialogue-based hospital administration simulation (Ayaz et al., 2021). Overall, H-ADMINSIM provides a standardized environment for systematic LLM evaluation in hospital administration and a simulation-based platform for assessing the feasibility of LLM-driven workflow automation.

2. Related Works

Most existing LLM-based healthcare simulation studies focus on clinical, physician-centered settings, where models are evaluated primarily on diagnostic accuracy and clinical reasoning through patient–physician interactions. Several works propose multi-agent frameworks in which multiple physician LLMs collaborate to improve clinical decision-making, typically benchmarked on medical question–answering datasets (Tang et al., 2024; Kim et al., 2024). In addition, task-level workflow simulations usually assign LLMs the role of physicians and evaluate diagnostic and clinical reasoning in dialogue-based clinical scenarios (Schmidgall et al., 2024; Liu et al., 2024; Liao et al., 2024; Fan et al., 2025). MedAgentSim (Almansoori et al., 2025) further introduces a self-improvement mechanism that iteratively enhances physician performance within such clinical simulations. LLM-based agent simulations have also been explored as educational tools, using role-playing with simulated patients to support clinical

training and communication skills (Li et al., 2024b; Wang et al., 2024; Wei et al., 2024; Hicke et al., 2025).

Among these simulation studies, only a few introduce limited administrative components. For example, Agent Hospital (Li et al., 2024a) and ClinicalLab (Yan et al., 2024) include a department guidance stage, where nurse-like agents assign patients to departments based on a single chief complaint or brief symptom descriptions. However, these components do not capture naturalistic intake workflows: they omit dialogue-based elicitation of patient symptoms and collection of patient demographics, and instead remain centered on patient–physician interactions and clinical reasoning. PIORS (Bao et al., 2024) proposes an outpatient administrative simulation focused on reception nurses, but its scope is restricted to conversational intake and department guidance, without modeling temporal constraints or scheduling workflows that are central to real-world operations. In contrast, our work introduces a hospital administration-centric simulation that integrates dialogue-based intake with temporally constrained scheduling workflows for realistic evaluation of outpatient administration.

3. Task Definition

This study focuses on first-visit outpatient workflows as a core subset of hospital administrative processes. A first-visit encounter is a canonical setting in which administrative staff directly elicit patient information and make foundational decisions under uncertainty, such as department assignment and appointment scheduling. In contrast, follow-up visits build on previous encounters and involve less extensive administrative intake, though they may require more complex scheduling for lab tests, imaging, or other procedures. Given this distinction, we restrict our scope to first-visit workflows, which serve as the entry point to outpatient care and the main stage for administrative decision-making.

Patient Intake. During patient intake, administrative staff collect demographic and medical information through dialogue-based interactions, including their chief complaint and, in higher-level hospitals, prior medical referrals. This information forms the basis for department assignment and appointment scheduling.

Appointment Scheduling. Once the patient’s department is determined, administrative staff elicit the patient’s scheduling preferences through dialogue and make an appointment accordingly. In higher-level hospitals, physician capacity—the number of outpa-

tients a physician sees per hour—varies by department and individual consultation style and is managed to prevent physician overburdening. Staff also handle rescheduling and cancellation requests from patients with existing appointments.

FHIR Integration. After completing these tasks, staff record patient demographics and appointment information in the hospital information system (HIS). However, HIS schemas differ across hospitals, leading to inconsistencies in data representation that hinder interoperability (Marfaglia et al., 2025). To mitigate this, many countries have adopted the FHIR standard for representing and exchanging medical data (Bossenko et al., 2024). In this study, we instantiate core FHIR resources to represent these administrative workflows.

4. H-AdminSim: Hospital Outpatient Administration Simulation

We present H-ADMINSIM, a simulation framework for first-visit outpatient administration in hospitals. The framework covers synthetic data generation, multi-agent simulation of administrative workflows, and integration with HIS via the FHIR standard.

4.1. Simulation Data Synthesis

H-ADMINSIM generates synthetic data following a hierarchical generation process with three stages: hospital, physician, and patient. Further details are provided in Appendix A.

4.1.1. HOSPITAL CONFIGURATIONS

Hospital configurations specify the time system as well as the hospital’s department information. The time system defines the hospital’s operating hours (*e.g.*, 10:00–17:00), time unit, and simulation period.

Time Unit. The time unit (τ) represents the minimum temporal interval in hospital operation modeling and serves as the foundation for all time-dependent features, including physician schedules and appointments. All time-related FHIR resources reference time slots generated based on τ . For example, an 8-hour hospital day with $\tau = 0.25$ hours (15 minutes) is divided into 32 time slots, with appointments and physician schedules managed accordingly.

Departments. This study focuses on nine internal medicine specialties that are widely recognized in clinical practice: gastroenterology, cardiology,

pulmonology, endocrinology/metabolism, nephrology, hematology/oncology, allergy, infectious diseases, and rheumatology.² Because internal medicine exhibits substantial symptom overlap, department assignment becomes particularly challenging and error-prone, underscoring the need for further study on assignment accuracy (Graber et al., 2005; Jain et al., 2014). In this stage, users can specify a range for the number of physicians per department, and the actual number is randomly determined within that range.

4.1.2. PHYSICIAN CONFIGURATIONS

Given the department-level physician counts defined in the hospital configuration, this stage generates physician profiles with randomly assigned demographics.

Working Days. In smaller hospitals, physicians typically provide outpatient services on all weekdays, whereas those in higher-level hospitals follow variable schedules, offering outpatient clinics only when they are not assigned to other duties such as inpatient care or research. To reflect these differences across hospital levels, the data generation process allows users to specify a range for the number of outpatient working days per physician within the simulation period.

Capacity. Physician capacity denotes the number of outpatients a physician can see per hour (**Physician Side Gigs**), with a theoretical upper bound of $1/\tau$, though practical capacity varies with each physician’s average consultation time. During data synthesis, users specify a capacity range, and each physician is randomly assigned a capacity within this range, capped by this upper bound.

Schedule. For each physician, two types of time slots—busy schedules and outpatient appointments—are prefilled on outpatient working days during data synthesis. The number of slots per appointment is determined by the physician’s capacity—for example, with $\tau = 0.25$ hours and a capacity of 2, each appointment occupies two consecutive slots (0.5 hours). Users can specify the daily proportions of busy schedules and appointments, enabling random generation of physician schedules within the defined bounds.

4.1.3. PATIENT CONFIGURATIONS

After prefilling virtual appointments for physicians, patient profiles are created to match the number of appointments, with demographic and condition-related attributes randomly assigned.

² References are provided in Appendix A.1.1

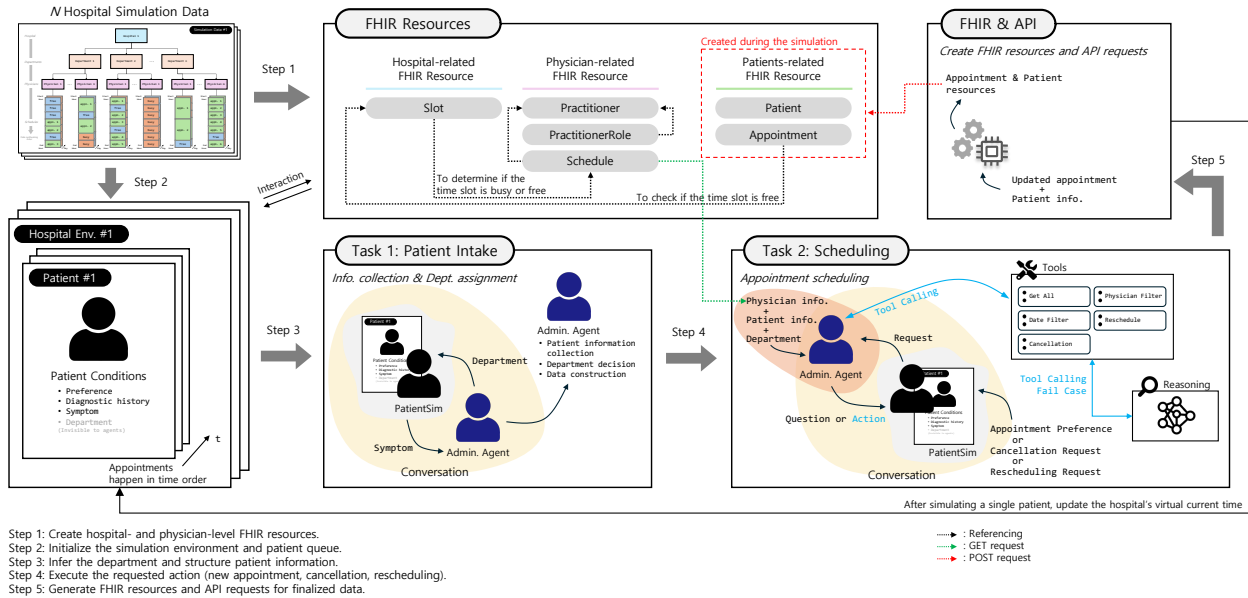


Figure 1: Diagram of the hospital administration simulation. Synthesized hospital data populate the hospital information system (HIS) with physician and hospital information using FHIR. The framework then simulates patient intake and appointment scheduling, uploading the resulting patient and appointment records to the HIS while keeping physician schedules updated via real-time FHIR communication. The hospital simulation data shown in the upper-left are shown in Appendix A.

Scheduling Preferences. Patient appointment scheduling preferences are categorized into three types that capture common patterns in real-world hospital administration: (1) earliest possible consultation (*asap*); (2) earliest consultation after a specified date (*date*); and (3) a preferred physician (*physician*) (Douglas et al., 2005; Patrick et al., 2008; Agrawal et al., 2023). In this study, these three preference types are combined into primary and secondary preferences, yielding six possible preference combinations per patient and reflecting the diverse ways in which preferences are expressed in administrative settings. Secondary preferences are probabilistically activated during the simulation.

Disease Profile. To construct patient profiles for simulation, each patient is assigned diseases and associated symptoms. Gold-standard labels specifying the correct target department for each disease are required to evaluate agent performance in department assignment. Symptom data for each disease were collected from the National Health Service (NHS) health encyclopedia,³ and department labels were created based on medical reference materials. From the 427

3. NHS Inform Health Encyclopedia Scotland

diseases listed by the NHS, 194 related to internal medicine were selected to build disease profiles that include department labels and symptom information across the nine specialties. Details of the labeling process are provided in Appendix A.2.

Diagnostic History. Patients are categorized as either (1) those without a prior diagnosis (*without_history*) or (2) those with a prior diagnosis (*with_history*). In primary and secondary care, patients generally visit without prior diagnoses, whereas in tertiary care many patients visit after receiving diagnoses at lower-level institutions and may prefer physicians specializing in their diagnosed condition (Physio-pedia, 2022; Keiser University, 2024). These diagnostic statuses are generated during data synthesis using predefined probabilities to simulate various hospital scenarios.

4.2. FHIR-Integrated Hospital Environment

FHIR Resources. After generating the simulation data, essential hospital information is uploaded to the HIS via FHIR to construct a realistic environment. Each physician is represented by *Practitioner* and *PractitionerRole* resources, and their working

days are encoded in a *Schedule* resource. *Slot* resources, segmented by the time unit (τ), are linked to each *Schedule* and indicate outpatient availability as *busy* or *free*, forming the reference timetable for agents. Patient information and scheduling outcomes are recorded in *Patient* and *Appointment* resources during task execution. Further details of the FHIR resources are provided in Appendix B.

Time Flow. Hospital administrative tasks are inherently time-dependent, making temporal flow a key component of the simulation. For instance, appointments cannot be scheduled in past time slots, and ongoing or completed appointments cannot be canceled. To capture such constraints, the simulation explicitly models temporal progression using a virtual current time, which can be initialized up to n days before the start of the simulation period. As time advances, appointment statuses are updated (*scheduled*, *in-progress*, *completed*), and rescheduling or cancellation is allowed only for *scheduled* appointments.

4.3. Simulation of Administration Tasks

As shown in Figure 1, H-ADMINSIM simulates hospital administrative workflows to emulate real-world outpatient administration, enabling systematic evaluation of administrative agents. The simulation environment comprises a patient agent and a staff agent responsible for administrative tasks.

4.3.1. PATIENT INTAKE

During the patient intake simulation, patient and administrative staff agents engage in a dialogue. The staff agent follows a sequence that begins with greeting, proceeds to collecting demographic information and asking about symptoms and prior diagnoses, and concludes with a department recommendation. Throughout the interaction, the staff agent maintains polite and empathetic communication while minimizing technical jargon (Kwame and Petrucka, 2021; Sharkiya, 2023). The patient agent’s persona is generated by integrating the synthesized patient profile—which includes scheduling preferences, prior diagnosis status, and disease–symptom information—with PATIENTSIM attributes (Kyung et al., 2025) such as personality, history recall level, confusion level, and language proficiency. Patients without prior diagnoses describe only their symptoms without referencing specific diseases. At the end of the dialogue, the staff agent extracts and structures the patient’s demographic information and department recommendation

for downstream tasks, and these outputs are evaluated using rubrics on data format correctness, simulation completeness, department appropriateness, and information extraction quality. Appendix D.1 provides details on PATIENTSIM attributes and prompts, and Appendix D.3 presents example simulation dialogues.

4.3.2. APPOINTMENT SCHEDULING

After patient intake, the staff agent elicits the patient’s scheduling preferences through dialogue and proposes an appointment based on physician availability in the assigned department and the current simulation time. When necessary, the agent asks clarifying questions until the preferences are fully identified, then queries the FHIR-based hospital environment to retrieve available outpatient slots. The patient agent first states a primary preference and may reject the proposed schedule with a 30% probability, in which case a secondary preference is provided. This design yields nine scheduling dialogue types, formed from three primary preference types and their six possible primary–secondary combinations. The staff agent identifies the patient’s preferences from the dialogue and, when it determines that a scheduling tool is needed, invokes the appropriate tool within the assigned department: (1) the earliest available slot across all physicians (*asap*); (2) the earliest slot after a specified date (*date*); or (3) the earliest slot for a preferred physician (*physician*). Tool-based scheduling (*Scheduling (T)*) requires tools that cover diverse patient preferences but yields stable performance. On the other hand, reasoning-based scheduling (*Scheduling (R)*) is prompt-based and easy to implement, yet performs substantially worse (up to 75.9 percentage points lower; Table 2). Consequently, our framework uses tool-based scheduling by default and falls back to reasoning-based scheduling only when no suitable tool is available or the request falls outside tool coverage. In this fallback mode, physician schedules retrieved from FHIR are included in the prompt, allowing the agent to infer the earliest feasible slot consistent with the patient’s preferences.

In addition to new appointment scheduling, the staff agent also manages existing patient schedules. When a patient inquiry arrives, the staff agent infers the patient’s intent through dialogue and invokes the appropriate tool for rescheduling (moving appointments earlier) or cancellation. In both cases, the agent first retrieves the original appointment; if retrieval fails due to missing information, the staff agent asks the

necessary details (*e.g.*, full name). For rescheduling, the appointment is reassigned if an eligible slot is available; otherwise, the patient is added to a waiting list. For cancellation, once the appointment is canceled and a slot becomes free, the staff agent iterates through the waiting list and automatically reassigns appointments to patients who can be moved forward.

Simulation outcomes are evaluated using rubrics that assess schedule validity, alignment with patient preferences, and the success of rescheduling and cancellation. Appendix D.2 provides prompts, and Appendix D.3 presents example simulation dialogues.

4.3.3. HOSPITAL INFORMATION SYSTEM UPDATE VIA FHIR

After all tasks are completed, new *Patient* and *Appointment* FHIR resources are created from the extracted patient information and final schedule. Schedule changes—rescheduling or cancellation—are handled by updating or deleting the relevant *Appointment* resources. All FHIR operations are executed in real time throughout the simulation.

5. Experiments

5.1. Experimental Setup

5.1.1. HOSPITAL-LEVEL DATA SYNTHESIS

Experiments were conducted on synthesized hospital datasets at three levels—primary, secondary, and tertiary—each reflecting distinct institutional characteristics. Higher-level hospitals include more departments and physicians and use finer time units, whereas primary hospitals involve patients with fewer prior diagnoses and weaker physician preferences during scheduling. For each level, three datasets were generated for a 7-day simulation, yielding 516, 769, and 5,052 synthesized patient profiles for primary, secondary, and tertiary hospitals, respectively.

Table 1 summarizes the key synthesis parameters. Among these, time granularity is an important design choice, as it directly affects scheduling resolution and physician capacity. For tertiary hospitals, the time unit was set to 0.05 hours (Irving et al., 2017)⁴ to reflect shorter outpatient consultations, whereas a coarser 0.25-hour unit was used for primary and secondary hospitals. Physician capacity was derived from the time unit and randomly assigned within

the allowable range for secondary and tertiary hospitals. Hospital scale was controlled mainly through the number of departments and physicians: all nine internal medicine specialties were included for tertiary hospitals, while primary and secondary hospitals were assigned fewer departments (2–3 and 7–9, respectively). Similarly, the number of physicians per department increased with the hospital level, ranging from one in primary hospitals to up to three in tertiary hospitals. Other synthesis parameters were tuned to reflect typical differences across hospital levels. In higher-level hospitals, patients were more likely to have prior diagnoses from lower-level institutions and to express explicit physician preferences. Conversely, primary hospitals have longer outpatient working days for physicians and more *asap* visits, with patients often arriving without prior appointments. Scheduling preference types were assigned hierarchically: the primary preference was first sampled according to the distribution in Table 1, and the secondary preference was then randomly selected from the remaining two types.

5.1.2. SIMULATION PARAMETERS

During the intake simulation, the dialogue between the patient and administrative staff remains brief, as it does not involve physician-led clinical consultation. Accordingly, each dialogue-based simulation (*e.g.*, intake and scheduling) is capped at five rounds, with each round consisting of a two-turn interaction between the patient and the staff. For intake simulations, the current round index is injected into the staff agent’s system prompt, enabling it to generate contextually appropriate questions and finalize the department in the last round. In the appointment scheduling simulation, rescheduling and cancellation event probabilities are set to 15% and 10% for tertiary hospitals—reflecting higher appointment demand and more tentative bookings—and 10% and 5% for primary and secondary hospitals, respectively.

5.2. Evaluation Rubrics

The simulation quality and task outcomes are evaluated using carefully designed rubrics. The rubrics consist of 10 evaluation criteria and 13 error codes, with some criteria associated with multiple error codes. Among them, 2 criteria and 2 error codes are shared between the intake and scheduling tasks. The intake task uses a total of 3 criteria and 5 error codes (Table 3), and the scheduling task uses 9 criteria and 10

4. Some countries report extremely short outpatient consultation times.

Table 1: Summary of the synthesized hospital dataset configuration for the 7-day simulation. For each hospital level, three hospital instances were randomly generated, and values in square brackets (\square) indicate sampling ranges. *Abbreviations: Dept. = department, h = hours, N = number, P = probability.*

Hospital Level	Time Unit (h)	N of Dept.	Physicians per Dept.	Working Days	Capacity	P of First Priority Preference (asap, physician, date)	P of Prior Diagnosis (w/o, w/)
Primary	0.25	[2, 3]	[1, 1]	[5, 7]	[4, 4]	(0.6, 0.2, 0.2)	(0.9, 0.1)
Secondary	0.25	[7, 9]	[1, 2]	[3, 4]	[1, 4]	(0.4, 0.4, 0.2)	(0.6, 0.4)
Tertiary	0.05	[9, 9]	[2, 3]	[3, 4]	[1, 20]	(0.4, 0.4, 0.2)	(0.2, 0.8)

Table 2: Success rate comparison across hospitals and models. *Scheduling (T)* uses tool-calling, while *Scheduling (R)* uses LLM reasoning only. Shaded cells indicate open-source LLM models. Bold and underlined values denote the best and second-best performance, respectively.

Hospital	Model	Task (%)		
		Intake	Scheduling (T)	Scheduling (R)
Primary	GPT-5 Mini	0.670±3.67	0.998±0.31	0.990±0.27
	GPT-5 Nano	0.791±6.61	0.967±0.52	0.556±10.04
	Gemini 2.5 Flash	0.889±2.77	0.998±0.22	0.371±5.59
	Llama 3.3 70B	0.818±5.34	0.941±0.98	0.402±7.35
	Qwen 3 32B	0.615±0.28	0.939±4.08	<u>0.594±3.16</u>
Secondary	GPT-5 Mini	0.570±1.34	<u>0.998±0.17</u>	0.967±1.73
	GPT-5 Nano	0.723±0.48	0.951±0.79	0.672±8.31
	Gemini 2.5 Flash	0.812±0.82	0.999±0.12	0.660±5.18
	Llama 3.3 70B	0.706±4.74	0.932±2.33	0.341±11.37
	Qwen 3 32B	0.528±4.14	0.963±1.34	<u>0.699±2.18</u>
Tertiary	GPT-5 Mini	0.613±0.79	<u>0.970±0.59</u>	0.750±1.45
	GPT-5 Nano	0.754±0.95	0.933±2.38	0.174±0.75
	Gemini 2.5 Flash	0.853±0.89	0.996±0.28	0.379±3.36
	Llama 3.3 70B	<u>0.768±0.77</u>	0.814±4.74	0.132±1.34
	Qwen 3 32B	0.576±0.46	0.929±1.07	0.238±3.67

error codes (Table 5). Among the intake criteria, one evaluates the patient agent, while the others evaluate the staff agent. The criteria are evaluated sequentially in a predefined order, and the evaluation terminates as soon as an error is detected, since several criteria depend on the outcomes of earlier ones. A task is deemed successful only if no error is triggered and all criteria in the sequence are satisfied. In particular, success in the intake task requires that no errors are triggered in either the patient or staff criteria. Detailed rubrics—including evaluation flow, error codes, and suspected causes—are provided in Appendix C.1.

5.3. Hospital-Level Simulation Results

H-ADMINSIM enables a systematic analysis of LLM-based hospital administration across hospital levels. We investigate how the hospital level affects task difficulty and the performance characteristics of LLMs.

Nine hospital datasets—three each for primary, secondary, and tertiary levels—were used for the ad-

ministrative simulation. Representative LLMs—GPT-5 Mini, GPT-5 Nano (OpenAI, 2025), Gemini 2.5 Flash (DeepMind, 2024), Llama 3.3 70B (Grattafiori et al., 2024), and Qwen 3 32B (Yang et al., 2025)—were evaluated, with the reasoning option set to *low* for the GPT-5 series, *dynamic* (default) for Gemini, and the default reasoning configuration for Qwen 3. Both the patient and staff agents were run with the same model to measure model-specific task success rates. Table 2 presents success rates for the patient intake and scheduling tasks, where the scheduling task includes rescheduling and cancellation events. The scheduling task was evaluated under two settings: *Scheduling (T)* (default), which prioritizes tool-calling for appointment scheduling and falls back to a reasoning-based mechanism upon failure, and *Scheduling (R)*, which performs the scheduling process without any tools (*i.e.*, purely through reasoning). Detailed analyses of the intake and scheduling results are provided in Sections 5.3.1 and 5.3.2, respectively. Lastly, the average token usage and API-related costs per simulation run are summarized in Appendix C.2.

5.3.1. INTAKE ANALYSIS

Hospital-level differences in intake performance are largely driven by whether patients have prior diagnoses and by the number of departments, both of which strongly influence department assignment accuracy. Other rubric criteria (*e.g.*, patient information extraction) are independent of hospital-level characteristics and instead depend on the model’s intrinsic capabilities. In Table 2, tertiary hospitals achieved higher intake success rates than secondary hospitals due to a larger proportion of patients with prior diagnoses. In contrast, primary hospitals achieved the highest success rates, even though they had the lowest proportion of patients referred with prior diagnoses. This is because the staff agent faced less ambiguity in department choices than in tertiary hospitals, which have up to four times more candidate departments.

Table 3: Error distribution in the intake task by hospital level, model, and error code. The prefixes of *Intake (Staff)* (e.g., *C1-1*, *C1-2*) denote the staff agent evaluation criteria and the order in which they are applied. Criterion *C1-2* can produce one of three error codes and *C2-1* of *Intake (Patient)* evaluates the completeness of the patient agent’s simulation (Appendix C.1). Shaded cells indicate open-source models. Bold and underlined values denote the best and second-best performance, respectively. *Abbreviations: IF = Incorrect Format, ID = Incorrect Department, IPI = Incorrect Patient Information, IDPI = Incorrect Department & Patient Information, IS = Incomplete Simulation.*

Hospital	Model	Intake (Staff)				Intake (Patient)	
		Errors	C1-1. IF	C1-2. ID	C1-2. IPI	C1-2. IDPI	C2-1. IS
Primary	GPT-5 Mini	61 / 516 (11.8%)	1 (1.6%)	59 (96.7%)	1 (1.6%)	-	127 / 516 (24.6%)
	GPT-5 Nano	88 / 516 (17.1%)	2 (2.3%)	85 (96.6%)	1 (1.1%)	-	33 / 516 (6.4%)
	Gemini 2.5 Flash	58 / 516 (11.2%)	-	58 (100.0%)	-	-	2 / 516 (0.4%)
	Llama 3.3 70B	96 / 516 (18.6%)	-	86 (89.6%)	6 (6.3%)	4 (4.2%)	14 / 516 (2.7%)
	Qwen 3 32B	106 / 516 (20.5%)	1 (0.9%)	100 (94.3%)	5 (4.7%)	-	120 / 516 (23.3%)
Secondary	GPT-5 Mini	152 / 769 (19.8%)	-	150 (98.7%)	2 (1.3%)	-	233 / 769 (30.3%)
	GPT-5 Nano	193 / 769 (25.1%)	2 (1.0%)	189 (97.9%)	1 (0.5%)	1 (0.5%)	27 / 769 (3.5%)
	Gemini 2.5 Flash	143 / 769 (18.6%)	-	142 (99.3%)	-	1 (0.7%)	2 / 769 (0.3%)
	Llama 3.3 70B	228 / 769 (29.7%)	-	218 (95.6%)	6 (2.6%)	4 (1.8%)	10 / 769 (1.3%)
	Qwen 3 32B	203 / 769 (26.4%)	1 (0.5%)	198 (97.5%)	3 (1.5%)	1 (0.5%)	207 / 769 (26.9%)
Tertiary	GPT-5 Mini	854 / 5,052 (16.9%)	2 (0.2%)	850 (99.5%)	2 (0.2%)	-	1,347 / 5,052 (26.7%)
	GPT-5 Nano	1,068 / 5,052 (21.1%)	10 (0.9%)	1,046 (97.9%)	9 (0.8%)	3 (0.3%)	231 / 5,052 (4.6%)
	Gemini 2.5 Flash	731 / 5,052 (14.5%)	-	728 (99.6%)	2 (0.3%)	1 (0.1%)	7 / 5,052 (0.1%)
	Llama 3.3 70B	1,170 / 5,052 (23.2%)	-	1,148 (98.1%)	14 (1.2%)	8 (0.7%)	23 / 5,052 (0.5%)
	Qwen 3 32B	992 / 5,052 (19.6%)	3 (0.3%)	961 (96.9%)	22 (2.2%)	6 (0.6%)	1,436 / 5,052 (28.4%)

Table 4: Department assignment error rates and average dialogue rounds (maximum five rounds) in the intake task across hospital levels and models. Shaded cells indicate open-source models. Bold and underlined values denote the best and second-best performance, respectively.

Hospital	Model	Avg. Rounds	Department Errors
Primary	GPT-5 Mini	4.86±0.44	59 / 516 (11.4%)
	GPT-5 Nano	<u>3.78±0.65</u>	85 / 516 (16.5%)
	Gemini 2.5 Flash	4.34±0.74	58 / 516 (11.2%)
	Llama 3.3 70B	3.86±0.36	90 / 516 (17.4%)
	Qwen 3 32B	2.78±0.67	101 / 516 (19.6%)
Secondary	GPT-5 Mini	4.75±0.60	150 / 769 (19.5%)
	GPT-5 Nano	<u>3.70±0.66</u>	191 / 769 (24.8%)
	Gemini 2.5 Flash	4.07±0.88	143 / 769 (18.6%)
	Llama 3.3 70B	3.71±0.46	222 / 769 (28.9%)
	Qwen 3 32B	2.67±0.70	200 / 769 (26.0%)
Tertiary	GPT-5 Mini	4.67±0.66	850 / 5,052 (16.8%)
	GPT-5 Nano	<u>3.54±0.70</u>	1,055 / 5,052 (20.9%)
	Gemini 2.5 Flash	3.62±0.82	729 / 5,052 (14.4%)
	Llama 3.3 70B	3.55±0.50	1,156 / 5,052 (22.9%)
	Qwen 3 32B	2.53±0.72	970 / 5,052 (19.2%)

Regarding model performance, Gemini 2.5 Flash achieved the highest success rates, whereas GPT-5 Mini underperformed relative to GPT-5 Nano. Among

open-source models, Llama 3.3 70B achieved performance comparable to commercial models, ranking second in primary and tertiary hospital simulations, while Qwen 3 30B showed the lowest performance across all settings. Table 3 shows the distribution of five intake error codes across three evaluation criteria. In the staff evaluation, all models frequently exhibited department assignment errors (*ID+IDPI*), whereas formatting errors (*IF*) and patient information extraction failures (*IPI+IDPI*) were rare. In the patient evaluation, GPT-5 Mini showed a substantially higher rate of patient simulation failures (*IS*) than other models, mainly due to the patient agent’s failure to provide demographic information to the staff. Although GPT-5 Mini achieved the second-best staff performance, its poorer patient performance led to a lower overall intake success rate (Table 2), as intake success requires both tasks to succeed.

Criterion *C1-1* in Table 3 assesses output formatting. If an error occurs at this stage, the intake task is classified as an *IF* error, and the evaluation does not proceed to *C1-2* (i.e., department and information extraction), regardless of whether the department assignment is correct or incorrect (Section 5.2). Therefore, we evaluate pure department assignment performance

Table 5: Error distribution in the tool-calling-based scheduling task by hospital level, model, and error code. The prefixes (e.g., *C3-1*, *C3-2*) denote the evaluation criteria and the order in which they are evaluated, where criterion *C3-7* can yield one of two error codes, and *C4-1* applies only to rescheduling and cancellation (Appendix C.1). Shaded cells indicate open-source models. Bold and underlined values denote the best and second-best performance, respectively. *Abbreviations: IS = Incomplete Simulation, IF = Incorrect Format, PC = Physician Conflict, IVS = Invalid Schedule, WD = Wrong Duration, TC = Time Conflict, IP = Incorrect Physician, IDT = Incorrect Date, NET = Not the Earliest Time, FI = Fail to Identify.*

Hospital	Model	Scheduling (T)										
		Errors	C3-1. IS	C3-2. IF	C3-3. PC	C3-4. IVS	C3-5. WD	C3-6. TC	C3-7. IP	C3-7. IDT	C3-8. NET	C4-1. FI
Primary	GPT-5 Mini	1 / 596 (0.2%)	-	-	-	-	-	-	-	-	1 (100.0%)	-
	GPT-5 Nano	20 / 590 (3.4%)	-	-	-	-	-	-	19 (95.0%)	1 (5.0%)	-	-
	Gemini 2.5 Flash	1 / 582 (0.2%)	-	-	-	-	-	-	-	-	-	1 (100.0%)
	Llama 3.3 70B	34 / 571 (6.0%)	-	1 (2.9%)	-	-	1 (2.9%)	-	4 (11.8%)	4 (11.8%)	24 (70.6%)	-
	Qwen 3 32B	30 / 577 (5.2%)	1 (3.3%)	-	-	-	1 (3.3%)	-	14 (46.7%)	7 (23.3%)	7 (23.3%)	-
Secondary	GPT-5 Mini	2 / 898 (0.2%)	-	-	-	-	-	-	-	2 (100.0%)	-	-
	GPT-5 Nano	46 / 908 (5.1%)	-	-	-	-	1 (2.2%)	-	1 (2.2%)	42 (91.3%)	2 (4.4%)	-
	Gemini 2.5 Flash	1 / 905 (0.1%)	-	-	-	-	-	-	-	-	-	1 (100.0%)
	Llama 3.3 70B	55 / 859 (6.4%)	-	-	-	-	-	-	-	13 (23.6%)	4 (7.3%)	38 (69.1%)
	Qwen 3 32B	28 / 858 (3.3%)	1 (3.6%)	-	-	-	1 (3.6%)	3 (10.7%)	3 (10.7%)	9 (32.1%)	11 (39.3%)	-
Tertiary	GPT-5 Mini	202 / 6,736 (3.0%)	1 (0.5%)	27 (13.4%)	-	-	14 (6.9%)	2 (1.0%)	6 (3.0%)	12 (5.9%)	140 (69.3%)	-
	GPT-5 Nano	455 / 6,811 (6.7%)	2 (0.4%)	81 (17.8%)	-	-	15 (3.3%)	19 (4.2%)	16 (3.5%)	229 (50.3%)	92 (20.2%)	1 (0.2%)
	Gemini 2.5 Flash	25 / 6,550 (0.4%)	-	-	-	-	-	-	-	-	23 (92.0%)	2 (8.0%)
	Llama 3.3 70B	1,174 / 6,441 (18.2%)	-	1 (0.1%)	-	4 (0.3%)	1 (0.1%)	28 (2.4%)	-	109 (9.3%)	167 (14.2%)	864 (73.6%)
	Qwen 3 32B	454 / 6,747 (6.7%)	2 (0.4%)	3 (0.7%)	-	-	-	62 (13.7%)	27 (6.0%)	81 (17.8%)	172 (37.9%)	107 (23.6%)

independently of the other rubric criteria, and report the results in Table 4. We additionally record the number of dialogue rounds (out of five) required for each model to determine the department in dialogue with the patient, as a measure of the decision-making efficiency of the staff agent. As this analysis ignores early termination in the rubric-based evaluation, department assignment errors previously masked by *IF* become observable, increasing error counts compared to Table 3 in some cases. Under this analysis, Gemini 2.5 Flash and GPT-5 Mini showed the fewest and second-fewest errors, respectively. Although GPT-5 Nano and Qwen 3 32B required the fewest rounds to reach decisions, their lower accuracy indicates an efficiency-accuracy trade-off. Decisions tended to occur earlier in higher-level hospitals, and accuracy was lowest in secondary hospitals and highest in primary hospitals, consistent with the overall intake success trends shown in Table 2. Further human evaluation of dialogue quality and department-level misclassification patterns is provided in Appendices C.3 and C.4.

5.3.2. SCHEDULING ANALYSIS

The scheduling task comprises three types of requests: new appointment scheduling, rescheduling of existing appointments, and appointment cancellation. Among these, rescheduling and cancellation require retrieval of the patient’s original appointment, making tool usage indispensable. However, new appointment

Table 6: Failure analysis in the *Scheduling (T)* new appointment task. *Wrong Tool* indicates failures due to incorrect tool invocation, and *Fallback Failures* refers to failures occurring during the reasoning-based scheduling process initiated by a fallback. Shaded cells indicate open-source models.

Hospital	Model	Wrong Tool	Fallback	
			Occurrences	Failures
Primary	GPT-5 Mini	0 / 516 (0.0%)	4 / 516 (0.8%)	1 / 4 (25.0%)
	GPT-5 Nano	19 / 516 (3.7%)	3 / 516 (0.6%)	1 / 3 (33.3%)
	Gemini 2.5 Flash	0 / 516 (0.0%)	0 / 516 (0.0%)	-
	Llama 3.3 70B	7 / 516 (1.4%)	4 / 516 (0.8%)	3 / 4 (75.0%)
	Qwen 3 32B	30 / 516 (5.8%)	7 / 516 (1.4%)	2 / 7 (28.6%)
Secondary	GPT-5 Mini	0 / 769 (0.0%)	6 / 769 (0.8%)	2 / 6 (33.3%)
	GPT-5 Nano	43 / 769 (5.6%)	3 / 769 (0.4%)	2 / 3 (66.7%)
	Gemini 2.5 Flash	0 / 769 (0.0%)	0 / 769 (0.0%)	-
	Llama 3.3 70B	15 / 769 (2.0%)	2 / 769 (0.3%)	2 / 2 (100.0%)
	Qwen 3 32B	21 / 769 (2.7%)	10 / 769 (1.3%)	6 / 10 (60.0%)
Tertiary	GPT-5 Mini	6 / 5,052 (0.1%)	47 / 5,052 (0.9%)	20 / 47 (42.6%)
	GPT-5 Nano	231 / 5,052 (4.6%)	19 / 5,052 (0.4%)	12 / 19 (63.2%)
	Gemini 2.5 Flash	0 / 5,052 (0.0%)	0 / 5,052 (0.0%)	-
	Llama 3.3 70B	236 / 5,052 (4.7%)	37 / 5,052 (0.7%)	22 / 37 (59.5%)
	Qwen 3 32B	144 / 5,052 (2.9%)	74 / 5,052 (1.5%)	29 / 74 (39.2%)

scheduling and rescheduling after retrieval can be performed either through tool-based execution or pure LLM reasoning, corresponding to the *Scheduling (T)* and *Scheduling (R)* settings in Table 2. Table 5 shows how the 10 scheduling error codes are distributed across nine evaluation criteria under the *Scheduling (T)* setting. The denominator of the *Scheduling (T)* task exceeds the number of patient profiles described in Section 5.1.1, because it incorporates both the

one-to-one new appointment scheduling attempts for each patient profile and additional scheduling errors produced during rescheduling and cancellation events.

In Table 2 under the *Scheduling (T)* setting, Gemini 2.5 Flash achieves the highest overall performance, followed by GPT-5 Mini, with both models exhibiting relatively stable performance across hospital levels. GPT-5 Nano, Llama 3.3 70B, and Qwen 3 32B also exhibited comparable performance levels. In Table 5, *FI* errors occur in rescheduling and cancellation requests when the original appointment cannot be retrieved, typically because the staff agent fails to elicit sufficient detail from the patient (*e.g.*, the physician’s full name). All other error codes are directly related to scheduling outcomes, and the causes of their occurrence can be categorized into two types: (1) cases where the agent misinterprets the patient’s intent and invokes an inappropriate tool (*Type I*); and (2) cases where the agent judges that the request “cannot be handled by tools” and falls back to reasoning-based scheduling, but the reasoning fails (*Type II*), corresponding to *Wrong Tool* and *Fallback Failures* in Table 6.

In Table 6, rescheduling and cancellation events, which occur randomly during the simulation, are excluded to ensure a fair comparison. Among the commercial models, GPT-5 Mini shows the highest rate of fallback transitions (*Fallback Occurrences*), most frequently judging the available tools to be insufficient, suggesting a conservative tendency in interpreting patient preference utterances. However, it exhibits the lowest failure rate among fallback cases, even though fallback failures are still observed. This trend is consistent with Table 2, where Mini achieves the lowest error rate under the *Scheduling (R)* (reasoning-based) setting. In other words, GPT-5 Mini’s errors in Table 5 are not primarily driven by incorrect tool selection (*Type I*), but reflect a small number of failures observed during frequent fallback reasoning (*Type II*). By contrast, in Table 6, GPT-5 Nano rarely judges that a request “cannot be handled by tools,” resulting in fewer fallback transitions than Mini. Instead, most of GPT-5 Nano’s errors arise from incorrect tool invocation (*Wrong Tool*), and even when fallback occurs, Nano shows a higher failure rate than GPT-5 Mini. This suggests that the majority of GPT-5 Nano’s scheduling errors in Table 5 are driven by wrong tool selection rather than failures in fallback reasoning (*Type I*). This shows that GPT-5 Nano has weaker recognition of patient scheduling preferences in utterances, resulting in more frequent incorrect tool selection. As shown in Table 6, Gemini 2.5 Flash neither

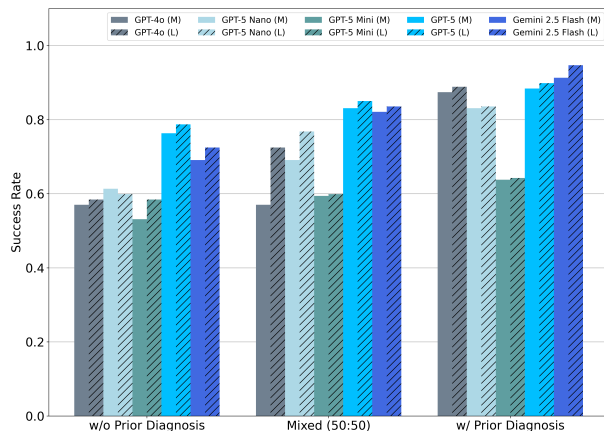


Figure 2: Comparison of intake task success rates across models under varying prior-diagnosis settings. Performance improves with higher prior-diagnosis rates and longer patient-staff dialogues (hatched bars).

invoked an incorrect tool nor triggered the fallback mechanism for new appointment requests, resulting in the most stable performance in the *Scheduling (T)*. For open-source models, most failures stem from incorrect tool selection (*Type I*) rather than fallback failures (*Type II*). When fallback is triggered, Qwen 3 32B exhibits a lower failure rate than Llama 3.3 70B, suggesting stronger reasoning-based scheduling performance, consistent with the results in Table 2 under the *Scheduling (R)* setting. Finally, detailed error analysis for the *Scheduling (R)* setting is provided in Appendix C.5.

5.4. Factors Affecting Intake Performance

While tool-based scheduling achieves high accuracy as long as appropriate tools are implemented (Table 2), patient intake remains the primary bottleneck for LLM-based hospital administration. As observed in Section 5.3.1, intake performance varies substantially across hospital levels depending on the availability of prior diagnoses and the model. To examine these effects more explicitly and explore conditions under which intake performance can be improved, we constructed three datasets: (1) all patients without prior diagnosis; (2) 50% with prior diagnosis; and (3) all with prior diagnosis, while keeping all other conditions identical. Each dataset includes patients across all nine internal medicine departments. Experiments were conducted using five models: GPT-4o (OpenAI, 2024), GPT-5 Nano, GPT-5 Mini, GPT-5, and Gemini

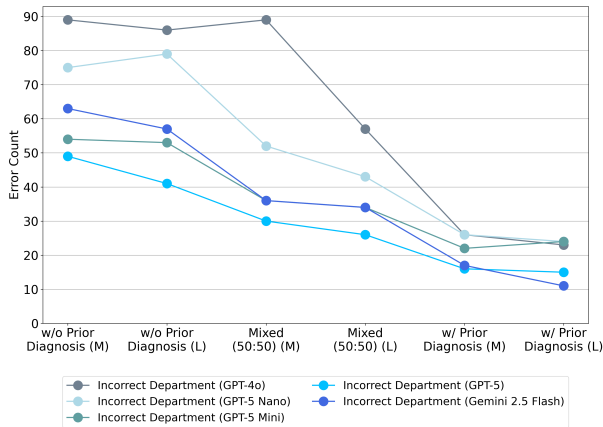


Figure 3: Department assignment errors under six conditions (three prior-diagnosis settings \times two conversation-round settings). Errors decrease with more patients having prior diagnoses and with longer conversations.

2.5 Flash. We also investigated the effect of conversation length by comparing simulations with maximum limits of five rounds (default) and eight rounds.

Figure 2 compares intake task success rates across models. For all models, success rates increased when patients had prior diagnoses, which support more accurate department recommendations. Longer patient-staff dialogues also slightly improved performance, suggesting that three additional dialogue rounds before department decision help refine symptom descriptions and enable additional clarification through follow-up questions. Figure 3 further analyzes errors in department assignment only, under six conditions that combine three prior-diagnosis settings and two conversation-round settings. The figure reports only department error counts, independent of the rubric evaluation order. The results indicate that both prior diagnoses and longer dialogues consistently reduce department assignment errors across models. In cases where the department had to be assigned based solely on symptoms (*i.e.*, without prior diagnoses), GPT-5 achieved the lowest department assignment error rates, whereas Gemini 2.5 Flash performed comparably when prior diagnoses were available. These findings suggest that model choice, enriching prior-diagnosis information, and allowing slightly longer intake dialogues are all effective levers for improving LLM-based intake performance.

6. Conclusion

This study presents H-ADMINSIM, a hospital administration simulation framework for synthetic data generation and end-to-end simulation of first-visit outpatient workflows with LLM agents. We construct detailed patient profiles using 194 labeled disease-symptom pairs across nine internal medicine specialties and model institution-specific characteristics in FHIR-integrated environments. H-ADMINSIM serves as a unified testbed for evaluating LLM-based administrative agents under diverse institutional settings. Our results show that tool-based scheduling achieves stable performance when appropriate tools are available, whereas patient intake remains the primary bottleneck. These findings suggest that, for reliable LLM-driven administration, robust tool-based scheduling pipelines are essential, and careful patient allocation for LLM-based intake is required.

Limitation and Ethical Discussion

Study Scope and Generalizability. This study focuses on simulating administrative workflows specifically for first-time outpatient visits. Future research should extend this framework to encompass diverse follow-up scenarios and more intricate administrative processes to ensure broader clinical applicability. **Safety and Data Privacy.** To ensure ethical rigor, all patient profiles and clinical data used in this study were synthetically generated, containing no real-world identifiable medical information. Thus, the dataset poses no risk of privacy infringement. However, practical deployment in clinical environments necessitates strict adherence to institutional data governance and privacy regulations. For API-based LLM services, robust safeguards—including data de-identification, stringent access control, and continuous auditing—must be implemented. Alternatively, deploying local LLMs within an on-premise hospital infrastructure can significantly mitigate external data leakage risks. Even when using models trained on real patient data, particularly when real-world patient data are collected via the simulator for training purposes, risk mitigation strategies such as differential privacy and output post-processing (*e.g.*, PII filtering) should be mandated to uphold the highest standards of patient confidentiality.

Acknowledgments

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants (No.RS-2019-III190075, No.RS-2025-02304967), the Korea Health Industry Development Institute (KHIDI) grant (No.RS-2025-02223382), and National Research Foundation of Korea (NRF) grant (NRF-2020H1D3A2A03100945, RS-2026-25484088), funded by the Korea government (MSIT, MOHW).

References

- Deepak Agrawal, Guodong Pang, and Soundar Kumara. Preference based scheduling in a healthcare provider network. *European Journal of Operational Research*, 307(3):1318–1335, 2023. ISSN 0377-2217. doi: <https://doi.org/10.1016/j.ejor.2022.09.027>. URL <https://www.sciencedirect.com/science/article/pii/S037722172200755X>.
- Mohammad Almansoori, Komal Kumar, and Hisham Cholakkal. Self-evolving multi-agent simulations for realistic clinical interactions, 2025. URL <https://arxiv.org/abs/2503.22678>.
- Muhammad Ayaz, Muhammad F Pasha, Mohammed Y Alzahrani, Rahmat Budiarto, and Deris Stiawan. The fast health interoperability resources (fhir) standard: Systematic literature review of implementations, applications, challenges and opportunities. *JMIR Med Inform*, 9(7):e21929, Jul 2021. ISSN 2291-9694. doi: 10.2196/21929. URL <https://medinform.jmir.org/2021/7/e21929>.
- Zhijie Bao, Qingyun Liu, Ying Guo, Zhengqiang Ye, Jun Shen, Shirong Xie, Jiajie Peng, Xuanjing Huang, and Zhongyu Wei. Piors: Personalized intelligent outpatient reception based on large language model with multi-agents medical scenario simulation, 2024. URL <https://arxiv.org/abs/2411.13902>.
- Duane Bender and Kamran Sartipi. Hl7 fhir: An agile and restful approach to healthcare information exchange. In *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, pages 326–331, 2013. doi: 10.1109/CBMS.2013.6627810.
- Alex M Bennett, Hannes Ulrich, Philip van Damme, Joshua Wiedekopf, and Alistair E W Johnson. Mimic-iv on fhir: converting a decade of in-patient data into an exchangeable, interoperable format. *Journal of the American Medical Informatics Association*, 30(4):718–725, 01 2023. ISSN 1527-974X. doi: 10.1093/jamia/ocad002. URL <https://doi.org/10.1093/jamia/ocad002>.
- Deepak Bhati, Meena S Deogade, and Deepika Kanyal. Improving patient outcomes through effective hospital administration: A comprehensive review. *Cureus*, 15(10):e47731, October 2023.
- Igor Bossenko, Rainer Randmaa, Gunnar Piho, and Peeter Ross. Interoperability of health data using fhir mapping language: transforming hl7 cda to fhir with reusable visual components. *Frontiers in Digital Health*, Volume 6 - 2024, 2024. ISSN 2673-253X. doi: 10.3389/fdgth.2024.1480600. URL <https://www.frontiersin.org/journals/digital-health/articles/10.3389/fdgth.2024.1480600>.
- Antonella Carbonaro, Luca Giorgetti, Lorenzo Ridolfi, Roberto Pasolini, Andrea Pagliarani, Martina Cavallucci, Alice Andalò, Livia Del Gaudio, Paolo De Angelis, Roberto Vespignani, and Nicola Gentili. From raw data to research-ready: A fhir-based transformation pipeline in a real-world oncology setting. *Computers in Biology and Medicine*, 197:111051, 2025. ISSN 0010-4825. doi: <https://doi.org/10.1016/j.combiomed.2025.111051>. URL <https://www.sciencedirect.com/science/article/pii/S0010482525014039>.
- Council of Europe. *Common European framework of reference for languages: Learning, teaching, assessment*. Cambridge University Press, 2001.
- Google DeepMind. Start building with gemini 2.5 flash, 2024. URL <https://developers.googleblog.com/en/start-building-with-gemini-25-flash/>. Accessed: 2025-10-05.
- KMJ Douglas, T Potter, GJ Treharne, K Obrenovic, ED Hale, A Pace, D Mitton, N Erb, A Whallett, JP Delamere, et al. Rheumatology patient preferences for timing and location of out-patient clinics. *Rheumatology*, 44(1):80–82, 2005.
- Zhihao Fan, Lai Wei, Jialong Tang, Wei Chen, Wang Siyuan, Zhongyu Wei, and Fei Huang. AI hospital: Benchmarking large language models in a multi-agent medical interaction simulator. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend

- Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert, editors, *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10183–10213, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL <https://aclanthology.org/2025.coling-main.680/>.
- Mark L Graber, Nancy Franklin, and Ruthanna Gordon. Diagnostic error in internal medicine. *Archives of internal medicine*, 165(13):1493–1499, 2005.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- Diwakar Gupta and Brian Denton. Appointment scheduling in health care: Challenges and opportunities. *IIE transactions*, 40(9):800–819, 2008.
- Yann Hicke, Jadon Geathers, Niroop Rajashekar, Colleen Chan, Anyanate Gwendolyne Jack, Justin Sewell, Mackenzi Preston, Susannah Cornes, Dennis Shung, and Rene Kizilcec. Medsimai: simulation and formative feedback generation to enhance deliberate practice in medical education. *arXiv preprint arXiv:2503.05793*, 2025.
- Greg Irving, Ana Luisa Neves, Hajira Dambha-Miller, Ai Oishi, Hiroko Tagashira, Anistasiya Verho, and John Holden. International variations in primary care physician consultation time: a systematic review of 67 countries. *BMJ Open*, 7(10):e017902, November 2017.
- Michael D Jain, George A Tomlinson, Danica Lam, Jessica Liu, Deepti Damaraju, Allan S Detsky, and Luke A Devine. Workplace-based assessment of internal medicine resident diagnostic accuracy. *Journal of graduate medical education*, 6(3):532–535, 2014.
- Yixing Jiang, Kameron C. Black, Gloria Geng, Danny Park, James Zou, Andrew Y. Ng, and Jonathan H. Chen. Medagentbench: A realistic virtual ehr environment to benchmark medical llm agents, 2025. URL <https://arxiv.org/abs/2501.14654>.
- Nobuyuki Kajiwara, Kazuyuki Hayashi, Masahiro Misago, Shinichiro Murakami, and Takato Ueoka. First-visit patients without a referral to the department of internal medicine at a medium-sized acute care hospital in japan: an observational study. *Int J Gen Med*, 10:335–345, October 2017.
- Keiser University. Primary, secondary, tertiary and quaternary: Understanding levels of patient care, 2024. URL <https://www.keiseruniversity.edu/primary-secondary-tertiary-and-quaternary-understanding-levels-of-patient-care/>.
- Yubin Kim, Chanwoo Park, Hyewon Jeong, Yik S Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Marzyeh Ghassemi, Cynthia Breazeal, and Hae W Park. Mdagents: An adaptive collaboration of llms for medical decision-making. *Advances in Neural Information Processing Systems*, 37:79410–79452, 2024.
- Abukari Kwame and Pammla M. Petrucka. A literature-based study of patient-centered care and communication in nurse-patient interactions: barriers, facilitators, and the way forward. *BMC Nursing*, 20(1), September 2021. ISSN 1472-6955. doi: 10.1186/s12912-021-00684-2. URL <http://dx.doi.org/10.1186/s12912-021-00684-2>.
- Daewon Kyung, Hyunseung Chung, Seongsu Bae, Jiho Kim, Jae Ho Sohn, Taerim Kim, Soo Kyung Kim, and Edward Choi. Patientsim: A persona-driven simulator for realistic doctor-patient interactions, 2025. URL <https://arxiv.org/abs/2505.17818>.
- Junkai Li, Yunghwei Lai, Weitao Li, Jingyi Ren, Meng Zhang, Xinhui Kang, Siyu Wang, Peng Li, Ya-Qin Zhang, Weizhi Ma, et al. Agent hospital: A simulacrum of hospital with evolvable medical agents. *arXiv preprint arXiv:2405.02957*, 2024a.
- Yanzeng Li, Cheng Zeng, Jialun Zhong, Ruoyu Zhang, Minhao Zhang, and Lei Zou. Leveraging large language model as simulated patients for clinical education, 2024b. URL <https://arxiv.org/abs/2404.13066>.
- Yusheng Liao, Yutong Meng, Yuhao Wang, Hongcheng Liu, Yanfeng Wang, and Yu Wang. Automatic interactive evaluation for large language models with state aware patient simulator, 2024. URL <https://arxiv.org/abs/2403.08495>.
- Kuan-Han Lin, Chun-Chi Hsu, and Kuan-Yin Lin. Job stress and burnout among hospital administrative staff: a cross-sectional study. *Scientific Reports*, 15(1):31064, Aug 2025. ISSN 2045-2322. doi: 10.1038/s41598-025-17209-5. URL <https://doi.org/10.1038/s41598-025-17209-5>.

- Ming-Hwai Lin, Hsiao-Ting Chang, Tzeng-Ji Chen, and Shinn-Jang Hwang. Why people select the outpatient clinic of medical centers: a nationwide analysis in taiwan. *PeerJ*, 8:e9829, August 2020.
- Ian Litchfield, Nicola Gale, Michael Burrows, and Sheila Greenfield. The future role of receptionists in primary care. *British Journal of General Practice*, 67(664):523–524, 2017. ISSN 0960-1643. doi: 10.3399/bjgp17X693401. URL <https://bjgp.org/content/67/664/523>.
- Hongcheng Liu, Yusheng Liao, Siqv Ou, Yuhao Wang, Heyang Liu, Yanfeng Wang, and Yu Wang. Med-pmc: Medical personalized multi-modal consultation with a proactive ask-first-observe-next paradigm, 2024. URL <https://arxiv.org/abs/2408.08693>.
- Alberto Marfoggia, Filippo Nardini, Valerio Antonio Arcobelli, Serena Moscato, Sabato Mellone, and Antonella Carbonaro. Towards real-world clinical data standardization: A modular fhir-driven transformation pipeline to enhance semantic interoperability in healthcare. *Computers in Biology and Medicine*, 187:109745, 2025. ISSN 0010-4825. doi: <https://doi.org/10.1016/j.compbimed.2025.109745>. URL <https://www.sciencedirect.com/science/article/pii/S0010482525000952>.
- Yeon Suk Oh, Mi Joon Lee, and Bum Jeun Seo. Relationship between job attitude and job performance of reception staff at general hospital in seoul. *Int J of Bio-Sci and Bio-Tech*, 7(3):155–162, 2019.
- OpenAI. Gpt-4o mini: advancing cost-efficient intelligence, 2024. URL <https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/>. Accessed: 2025-10-05.
- OpenAI. Introducing gpt-5, 2025. URL <https://openai.com/index/introducing-gpt-5/>. Accessed: 2025-10-05.
- Jonathan Patrick, Martin L Puterman, and Maurice Queyranne. Dynamic multipriority patient scheduling for a diagnostic resource. *Operations research*, 56(6):1507–1525, 2008.
- Physician Side Gigs. Average number of patients seen per hour, by physician specialty. URL <https://www.physiciansidegigs.com/average-number-of-patients-seen-per-hour>. Accessed: 2025-10-28.
- Physiopedia. Levels of healthcare, 2022. URL https://www.physio-pedia.com/index.php?title=Levels_of_Healthcare&oldid=308271. Accessed: 2025-10-28.
- Samuel Schmidgall, Rojin Ziaei, Carl Harris, Eduardo Reis, Jeffrey Jopling, and Michael Moor. Agent-clinic: a multimodal agent benchmark to evaluate ai in simulated clinical environments, 2024.
- Samer H. Sharkiya. Quality communication can improve patient-centred health outcomes among older patients: a rapid review. *BMC Health Services Research*, 23(1), August 2023. ISSN 1472-6963. doi: 10.1186/s12913-023-09869-8. URL <http://dx.doi.org/10.1186/s12913-023-09869-8>.
- Xiangru Tang, Anni Zou, Zhuosheng Zhang, Ziming Li, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. Medagents: Large language models as collaborators for zero-shot medical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 599–621, 2024.
- Peixing Wan, Zigeng Huang, Wenjun Tang, Yulan Nie, Dajun Pei, Shaofen Deng, Jing Chen, Yizhi Zhou, Hongru Duan, Qingyu Chen, and Erping Long. Outpatient reception via collaboration between nurses and a large language model: a randomized controlled trial. *Nature Medicine*, July 2024. ISSN 1078-8956, 1546-170X. doi: 10.1038/s41591-024-03148-7. URL <https://www.nature.com/articles/s41591-024-03148-7>.
- Ruiyi Wang, Stephanie Milani, Jamie C. Chiu, Jiayin Zhi, Shaun M. Eack, Travis Labrum, Samuel M Murphy, Nev Jones, Kate V Hardy, Hong Shen, Fei Fang, and Zhiyu Chen. PATIENT- ψ : Using large language models to simulate patients for training mental health professionals. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12772–12797, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.711. URL <https://aclanthology.org/2024.emnlp-main.711/>.
- Hao Wei, Jianing Qiu, Haibao Yu, and Wu Yuan. Medco: Medical education copilots based on a multi-agent framework. In *European Conference on Computer Vision*, pages 119–135. Springer, 2024.

Weixiang Yan, Haitian Liu, Tengxiao Wu, Qian Chen, Wen Wang, Haoyuan Chai, Jiayi Wang, Weishan Zhao, Yixin Zhang, Renjun Zhang, et al. Clinicalab: Aligning agents for multi-departmental clinical diagnostics in the real world. *arXiv preprint arXiv:2406.13890*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiayi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL <https://arxiv.org/abs/2505.09388>.

Algorithm 1 Synthetic data generation process

Input: Hospital H_i (size n_h); Department D_{ij} ($n_d \in [m_d, M_d]$); Physician PH_{ijk} ($n_{ph} \in [m_{ph}, M_{ph}]$); Patient PT_{ijkl} ($n_{pt} \in [m_{pt}, M_{pt}]$)

```

for  $i \leftarrow 1$  to  $n_h$  do
  Initialize  $H_i$  (e.g., time system)
   $n_d \leftarrow \text{random}(m_d, M_d)$ 
  for  $j \leftarrow 1$  to  $n_d$  do
     $n_{ph} \leftarrow \text{random}(m_{ph}, M_{ph})$ 
    for  $k \leftarrow 1$  to  $n_{ph}$  do
      Generate physician profile  $PH_{ijk}$ 
       $n_{pt} \leftarrow \text{random}(m_{pt}, M_{pt})$ 
      for  $l \leftarrow 1$  to  $n_{pt}$  do
        | Generate patient profile  $PT_{ijkl}$ 
      end
    end
  end
end

```

Appendix A. Data Synthesis Details

A.1. Data Synthesis Process

The data synthesis procedure for the hospital administration simulation follows a hierarchical generation process with three stages (Algorithm 1): hospital, physician, patient. An overview of the hierarchical structure of the data synthesis procedure is illustrated in Figure 4.

A.1.1. HOSPITAL DATA

The synthesis of simulation data begins with generating hospital- and department-level information. In this stage, users can specify parameters such as the number of hospitals to simulate, simulation period, start date, time unit (τ), and operating hours. A single τ corresponds to one discrete time interval (or time slot) within the hospital’s temporal system. In addition, the number of departments and the number of physicians assigned to each department can be configured. An example configuration is provided in the *Hospital* section of Table 7.

Departments. We selected nine internal medicine specialties that are common across the specialty lists recognized by leading medical organizations in each country—the American College of Osteopathic Internists (United States),⁵ the General Medical Council (United Kingdom),⁶ the Medical Board of Australia (Ahpra),⁷ and the Canadian Medical

5. American College of Osteopathic Internists

6. General Medical Council

7. Medical Board of Australia

Association (CMA)⁸ : gastroenterology, cardiology, pulmonology, endocrinology/metabolism, nephrology, hematology/oncology, allergy, infectious diseases, and rheumatology.

A.1.2. PHYSICIAN DATA

After hospital and department information are generated, physician data are synthesized according to the predefined number of physicians per department. Demographic attributes—including name, gender, birth date, and contact information (referred to as *telecom* in FHIR)—are first created, followed by operational details such as working days, capacity, and schedules. Working days are defined as days on which physicians are available for outpatient consultations; all time slots on working days are initialized as *free*, whereas those on non-working days are marked as *busy*. The number of working days must not exceed the length of the simulation period. Physician capacity denotes the number of patients that can be consulted per hour, with the maximum capacity defined as $1/\tau$, where τ is the hospital time unit. The actual capacity is randomly sampled from the divisors of this maximum, constrained by a predefined range. Schedule synthesis incorporates probabilistic parameters. In Table 7, the parameter *busy_schedule_prob* determines the likelihood that at least one *busy* slot appears on a working day, and the proportion of *busy* slots is sampled from the interval specified by *busy_schedule_ratio*. A subset of the remaining *free* slots is then assigned to appointments based on the *appointment_ratio*. The size of an appointment block is determined by the physician’s capacity, resulting in varying slot durations across physicians. Examples of physician-level synthesis parameters are provided in the *Physician* section of Table 7.

A.1.3. PATIENT DATA

After the total number of appointments is determined, patient profiles are generated and assigned to departments and physicians. Demographic attributes—including name, gender, birth date, contact information (referred to as *telecom* in FHIR), identifier, and address—are randomly synthesized. The identifier corresponds to a unique numeric value (*e.g.*, a national identification number) that distinguishes each patient. Patient scheduling preferences are assigned probabilistically among three types: *asap*, *physician*,

8. Canadian Medical Association

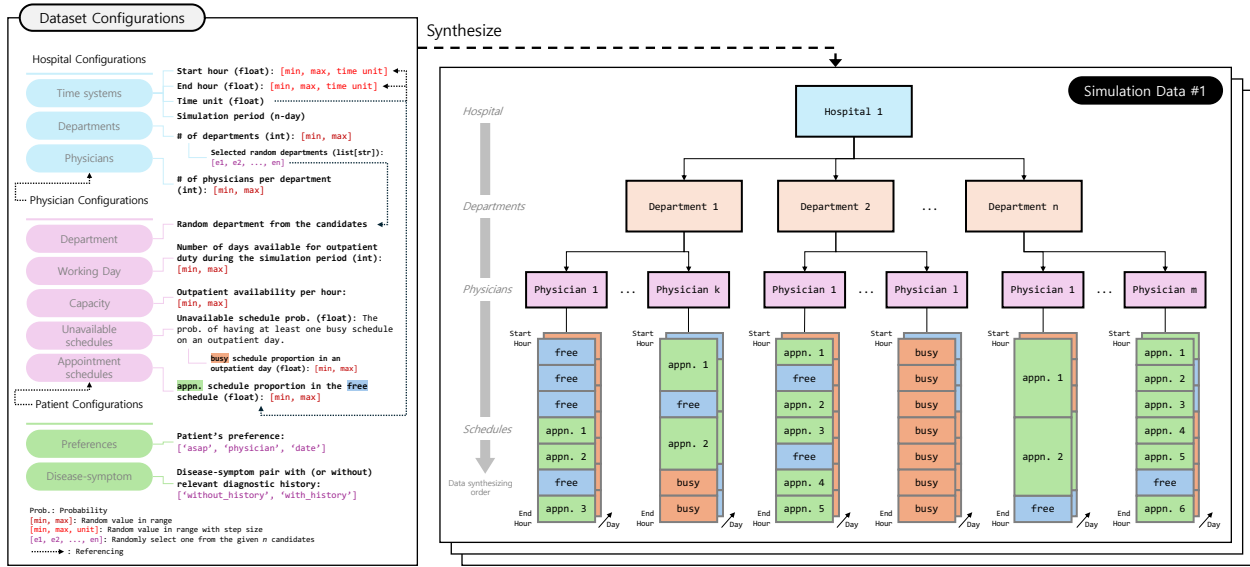


Figure 4: Overview of the hierarchical structure of the data synthesis procedure. The left panel presents the configuration parameters that govern data generation, and the right panel illustrates the hierarchical synthesis of hospital, department, physician, and patient data. Physician schedules consist of *busy* and *free* time slots, where *free* slots represent periods available for outpatient appointments, and a subset of these *free* slots is allocated as appointment blocks.

and *date* (Douglas et al., 2005; Patrick et al., 2008; Agrawal et al., 2023). The *asap* preference denotes scheduling the earliest available appointment with any physician within the designated department; the *physician* preference specifies scheduling with a particular physician at their earliest availability; and the *date* preference schedules the earliest appointment after a specified date. These three preference types define primary and secondary scheduling preferences, where the *preference* parameter in Table 7 determines the primary type and the secondary type is randomly selected from the remaining two.

Each patient is also randomly assigned a disease–symptom profile consistent with their department. The presence of a prior diagnosis is assigned probabilistically according to the synthesis configuration. Patients without a prior diagnosis are assumed to be unaware of their condition during the intake simulation. Relevant synthesis parameters are summarized in the *Patient* section of Table 7, and the construction of disease–symptom pairs and their associated department labels is described in Appendix A.2.

A.2. Construction of Disease–Symptom Datasets with Golden Department Labels

A.2.1. OBTAINING DISEASE AND SYMPTOM PAIRS

To simulate outpatient intake scenarios, disease–symptom profiles were constructed for patient profiles. We used the Scotland National Health Service (NHS) website,⁹ which provides disease-centered information through A-to-Z health encyclopedias maintained by various national NHS entities. Through web crawling, we collected 427 disease–symptom pairs, of which 194 corresponded to the nine internal medicine departments targeted in this study. The crawled data were subsequently post-processed using a Large Language Model (LLM) to extract symptom descriptions, yielding the final disease–symptom dataset. GPT-4o (OpenAI, 2024) was used for post-processing due to its efficiency, and the prompts employed for this step are illustrated in Figure 5.

Table 7: Synthesized data specifications for hospital, physician, and patient entities. Italicized variables indicate parameters that are randomly assigned or generated during data synthesis.

Entity Category	Variables (YAML Parameter)	Variable Examples of YAML Format
Hospital	<ul style="list-style-type: none"> Number of hospitals (<i>hospital_n</i>) Simulation start date (<i>start_date</i>) Simulation periods (<i>days</i>) Time unit (<i>time_unit</i>) Hospital start hour (<i>start_hour</i>) Hospital end hour (<i>end_hour</i>) The number of departments per hospital (<i>departemnt_per_hospital</i>) The number of physicians per department (<i>physician_per_department</i>) 	<pre>hospital_n : 10 start_date : min : 2025-03-17 max : 2025-09-21 days : 7 time_unit : 0.25 start_hour : min : 9 max : 10 end_hour : min : 18 max : 19 department_per_hospital : min : 7 max : 9 physician_per_department : min : 2 max : 3</pre>
Physician	<ul style="list-style-type: none"> <i>Name</i> <i>Gender</i> <i>Birth date</i> <i>Telecom</i> Working days (<i>working_days</i>) Capacity per hour (<i>capacity_per_hour</i>) Unavailable schedule probability (<i>busy_schedule_prob</i>) Unavailable schedule ratio (<i>busy_schedule_ratio</i>) Appointment ratio (<i>appointment_ratio</i>) 	<pre>working_days : min : 3 max : 4 capacity_per_hour : min : 1 max : 4 busy_schedule_prob : 0 busy_schedule_ratio : min : 0.4 max : 0.6 appointment_ratio : min : 0.2 max : 0.5</pre>
Patient	<ul style="list-style-type: none"> <i>Name</i> <i>Gender</i> <i>Birth date</i> <i>Telecom</i> <i>Identifier</i> <i>Address</i> Preference (<i>preference</i>) Symptom level (<i>symtom</i>) 	<pre>preference : type : [asap, physician, date] probs : [0.4, 0.4, 0.2] symptom : type : [without_history, with_history] probs : [0.7, 0.3]</pre>

A.2.2. GOLDEN DEPARTMENT LABELING

After constructing the disease–symptom dataset, golden labels were defined to indicate the medical departments responsible for treating each disease, serving as the reference standard for evaluating department assignment performance in the intake simulation. All diseases were manually annotated using three authoritative publicly available clinical information sources: Seoul National University Hospital Medicine Information,¹⁰ the Asan Medical Center Disease Ency-

clopedia,¹¹ and the Severance Hospital Disease Encyclopedia.¹² Each disease was mapped to one or more of the nine internal medicine specialties. Summary statistics and representative examples are provided in Table 8. The total disease count reported in Table 8 exceeds 194 because diseases treatable across multiple departments are counted once for each applicable specialty.

9. [NHS Inform Health Encyclopedia Scotland](#).
 10. [Seoul National University Hospital Medicine Information](#)

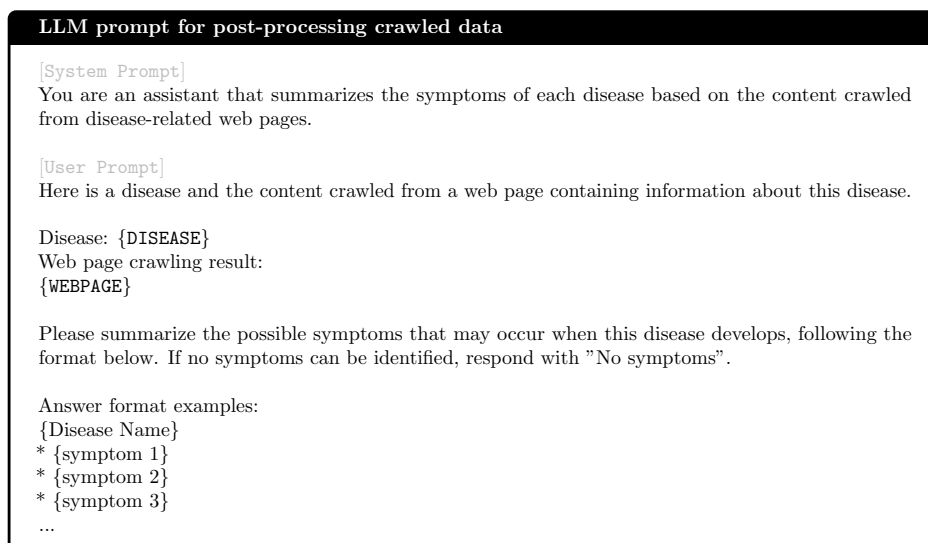


Figure 5: System and user prompts used by the LLM for post-processing crawled data and extracting structured disease–symptom pairs. {DISEASE} and {WEBPAGE} denote placeholders for the disease name and its corresponding crawled webpage, respectively.

A.3. Examples of Synthetic Data

As illustrated in Figure 6, the synthesized dataset for the simulation comprises metadata describing the simulation configuration, department information, physician profiles, and patient profiles with associated constraints and preferences. The data are represented in JSON format, which preserves structured relationships and referential links among entities.

Appendix B. FHIR Resources

We adopt Fast Healthcare Interoperability Resources (FHIR) R5 (Bender and Sartipi, 2013), the latest version developed by Health Level Seven International.¹³ FHIR is a standardized framework for representing and exchanging healthcare information, enabling interoperability of electronic health records across institutions. By leveraging web technologies (*i.e.*, RESTful APIs) and supporting JSON and XML formats, FHIR facilitates efficient and structured data exchange. Accordingly, we integrate the FHIR standard into our simulation and summarize the key resources used in this work, with representative FHIR resource instances illustrated in Figure 7.

11. [Asan Medical Center Disease Encyclopedia](#)
12. [Severance Hospital Disease Encyclopedia](#)
13. [The official website of HL7 International](#)

Practitioner. The *Practitioner* resource represents individuals who provide healthcare or related services and contains demographic attributes such as name, gender, birth date, and contact information. In the simulation, this resource is used to represent each physician’s demographic profile.

PractitionerRole. The *PractitionerRole* resource captures professional information about a practitioner, including workplace, specialty, and role-specific characteristics. Here, it is used to record attributes such as department affiliation and outpatient capacity, which are essential for retrieving physician schedules and assigning appointments.

Patient. The *Patient* resource stores demographic and administrative information about individuals receiving care, including name, gender, birth date, identifiers, and contact details. This resource represents each simulated patient within the system.

Schedule. The *Schedule* resource defines a practitioner’s general availability across multiple time periods. Each schedule links to a *Practitioner* and specifies the overall availability window without exposing individual appointment details, ensuring privacy. In our simulation, one schedule is created per practitioner for the entire simulation period.

Slot. A *Slot* subdivides the *Schedule* into discrete time intervals defined by the simulation time unit (τ). Each slot references its parent *Schedule* resource and

Table 8: Golden labels for disease–symptom pairs, listing the responsible departments, the number of diseases per department, and representative examples. Italicized disease names indicate cases in which treatment is possible across multiple departments.

Department	Disease Count	Disease Examples
Gastroenterology	35	Escherichia coli (E. coli) O157, <i>Gastroenteritis</i> , Coeliac disease, Stomach ulcer, Norovirus, Inflammatory bowel disease (IBD), Hepatitis B, Food poisoning, etc.
Cardiology	19	Abdominal aortic aneurysm, Angina, Arterial thrombosis, Atrial fibrillation, Heart attack, Heart block, Supraventricular tachycardia, etc.
Pulmonology	27	Sepsis, Asbestosis, <i>Asthma</i> , Breathlessness, Bronchiectasis, Catarrh, <i>Chest infection</i> , <i>Cystic fibrosis</i> , <i>Carcinoid syndrome and carcinoid tumours</i> , etc.
Endocrinology/Metabolism	15	Addison’s disease, <i>Cystic fibrosis</i> , Dehydration, <i>Gastroenteritis</i> , High blood sugar, Obesity, Osteoporosis, Overactive thyroid, Diabetes (Type 1, Type 2), etc.
Nephrology	5	Chronic kidney disease, <i>Kidney cancer</i> , <i>Kidney infection</i> , Kidney stones, Urinary tract infection (UTI)
Hematology/Oncology	49	Acute lymphoblastic leukaemia, Aplastic anaemia, <i>Carcinoid syndrome and carcinoid tumours</i> , Ewing sarcoma, <i>Kidney cancer</i> , Mesothelioma, etc.
Infectious diseases	42	Clostridium difficile, <i>Chest infection</i> , <i>Croup</i> , Ebola virus disease, <i>Kidney infection</i> , Threadworms, Thrush, Tick bites, Strep A, etc.
Allergy	18	<i>Asthma</i> , Food allergy, Hay fever, Hives, Allergic rhinitis, Allergies, etc.
Rheumatology	18	<i>Ankylosing spondylitis</i> , <i>Arthritis</i> , <i>Costochondritis</i> , <i>Fibromyalgia</i> , <i>Gout</i> , <i>Lupus</i> , etc.

indicates whether the practitioner is available or busy during that interval. For example, with $\tau = 0.25$ hours, each slot corresponds to a 15-minute unit. In this study, slots are marked as either *free* or *busy*, indicating whether the interval is available for appointment booking or already occupied. This structure is essential for constructing the practitioner’s complete schedule throughout the simulation.

Appointment. The *Appointment* resource represents an outpatient encounter between a patient and a physician. Appointments reference one or more consecutive *free Slot* resources (depending on physician capacity) and link to both the *Practitioner* and *Patient* resources. During the simulation, appointment creation, cancellation, and modification update physician schedules in real time.

Appendix C. Experimental Details

C.1. Evaluation Rubrics

The proposed framework evaluates task outcomes using predefined rubrics. Each task rubric includes a set of criteria and associated error codes: the intake task uses three criteria with three task-specific error codes and two common error codes, while the scheduling task uses eight criteria with seven task-specific error codes and two common error codes. Rescheduling and cancellation events each include one task-specific error code. Among the intake criteria, one evaluates the patient agent, while the others evaluate the staff agent. The criteria are evaluated sequentially in a predefined order, and the evaluation terminates as soon as an error is detected, since several criteria depend on the outcomes of earlier ones. A task is deemed successful only if no error is triggered and all criteria in the sequence are satisfied. Figure 8 shows the overall evaluation flow.

```

Example of synthesized data for the simulation

{
  "metadata": {
    "hospital_name": "hospital_00",
    "start_date": "2025-04-17",
    "end_date": "2025-04-18",
    "days": 2,
    "department_num": 1,
    "doctor_num": 1,
    "time": {"start_hour": 9.0, "end_hour": 18.0, "time_unit": 0.25}
  },
  "department": {
    "endocrinology/metabolism": {"code": "IMEND", "doctor": ["Dr. Benedict Tomerlin"]}
  },
  "doctor": {
    "Dr. Benedict Tomerlin": {
      "department": "endocrinology/metabolism",
      "specialty": {"name": "Osteoporosis and Metabolic Bone Disease", "code": "IMEND-2"},
      "schedule": {"2025-04-17": [], "2025-04-18": [[9.0, 18.0]]},
      "capacity_per_hour": 4,
      "capacity": 36,
      "gender": "female",
      "telecom": [{"system": "phone", "value": "+82844559851970", "use": "work"}],
      "birthDate": "1961-03-21"
    }
  },
  "patient": [{
    "patient": "Reynaldo Verlotte",
    "gender": "female",
    "telecom": [{"system": "phone", "value": "+8275129708711", "use": "work"}],
    "birthDate": "1982-09-21",
    "identifier": [{"value": "820921-1133985", "use": "official"}],
    "address": [{"type": "postal", "text": "0, Hoedong-ro, Yecheon-gun, Gunsan-si", "use": "home"}],
    "constraint": {
      "preference": ["physician", "asap"],
      "attending_physician": "Dr. Benedict Tomerlin",
      "valid_from": "N/A",
      "symptom_level": "without_history",
      "symptom": {
        "disease": "Dehydration",
        "department": ["endocrinology/metabolism"],
        "symptom": ["Peeing less often than usual", "Headache"]
      }
    }
  }
]}

```

Figure 6: Example of synthesized data used in the outpatient administration simulation, illustrating hospital (metadata), department information, and physician and patient profiles in JSON format.

Regarding the intake task, the framework assesses both the patient and the staff agents. The staff evaluation focuses on adherence to the output format and the accuracy of information extraction and department assignment, while the patient evaluation ensures that the simulation is complete—*i.e.*, whether the agent faithfully conveys the pre-assigned role-play information. These two evaluations are performed independently, and the intake task is considered successful only if both evaluations are passed. For the scheduling task, the evaluation examines whether the staff agent completes the task execution, and then assesses response formatting, the validity of the generated schedule, the appropriateness of the assigned physician, and whether patient preferences are properly reflected. The framework also verifies whether the agent identifies the earliest feasible appointment

slot while satisfying all scheduling constraints. For the rescheduling and cancellation events, the rubric first checks whether the agent successfully retrieves the patient’s requested original schedule and then follows different evaluation flows depending on the event type. The full mapping between error codes and their corresponding suspected causes is provided in Table 9.

C.2. Token Usage and Operational Costs

This section analyzes token usage and the associated economic cost in simulations using API-based LLMs. Table 10 reports the average token consumption per simulation run and the corresponding API cost, aggregated over 50 runs for each task and model. Since rescheduling and cancellation events occur randomly and vary with the length of the waiting list, we re-

Example of FHIR resources

```

===== Practitioner =====
{
  "resourceType": "Practitioner",
  "id": "hospital01-imrhe-Dr.LincolnBendzus",
  "active": true,
  "name": [{"family": "Bendzus", "given": ["Lincoln"], "prefix": ["Dr."]},],
  "telecom": [{"system": "phone", "value": "+825959635756519", "use": "work"}],
  "gender": "male",
  "birthDate": "1960-03-30"
}
===== PractitionerRole =====
{
  "resourceType": "PractitionerRole",
  "id": "hospital01-imall-Dr.ThurmanModdejonge-role",
  "active": true,
  "practitioner": [{"reference": "Practitioner/hospital01-imall-Dr.ThurmanModdejonge"}],
  "specialty": [
    {"coding": [{"code": "IMALL-2", "display": "Food and Drug Allergy"}], "text": "allergy"}
  ],
  "characteristic": [
    {"coding": [{"code": "capacity_per_hour", "display": "4"}], "text": "capacity_per_hour"},
    {"coding": [{"code": "capacity", "display": "192"}], "text": "capacity"}
  ]
}
===== Schedule =====
{
  "resourceType": "Schedule",
  "id": "hospital01-imrhe-Dr.LincolnBendzus-schedule",
  "active": true,
  "actor": [{"reference": "Practitioner/hospital01-imrhe-Dr.LincolnBendzus"}],
  "planningHorizon": {"start": "2025-03-22T10:00:00+09:00", "end": "2025-03-28T18:00:00+09:00"}
}
===== Slot =====
{
  "resourceType": "Slot",
  "id": "hospital01-imrhe-Dr.LincolnBendzus-20250325-slot10",
  "schedule": {"reference": "Schedule/hospital01-imrhe-Dr.LincolnBendzus-schedule"},
  "status": "busy",
  "start": "2025-03-25T12:30:00+09:00",
  "end": "2025-03-25T12:45:00+09:00"
}
===== Patient =====
{
  "resourceType": "Patient",
  "id": "hospital02-iminf-EarnestMetoxen",
  "identifier": [{"use": "official", "value": "961010-7363710"}],
  "active": true,
  "name": [{"family": "Metoxen", "given": ["Earnest"]}],
  "telecom": [{"system": "phone", "value": "+82693618062105", "use": "mobile"}],
  "gender": "male",
  "birthDate": "1996-10-10",
  "address": [{"use": "home", "type": "postal", "text": "4, Dongbang, Geumsan-gun, Hanam-si"}]
}
===== Appointment =====
{
  "resourceType": "Appointment",
  "id": "hospital02-iminf-Dr.AldenGaestel-20250918-appn0-0",
  "status": "booked",
  "start": "2025-09-18T10:00:00+09:00",
  "end": "2025-09-18T10:15:00+09:00",
  "slot": [{"reference": "Slot/hospital02-iminf-Dr.AldenGaestel-20250918-slot0"}],
  "participant": [
    {"actor": {
      "reference": "Practitioner/hospital02-iminf-Dr.AldenGaestel",
      "display": "Dr. Alden Gaestel"
    }, "status": "accepted"},
    {"actor": {
      "reference": "Patient/hospital02-iminf-EarnestMetoxen",
      "display": "Earnest Metoxen"
    }, "status": "accepted"}
  ]
}

```

Figure 7: Example FHIR resource instances used in the simulation.

strict our analysis to the new appointment scheduling scenario to ensure consistency in resource usage measurements.

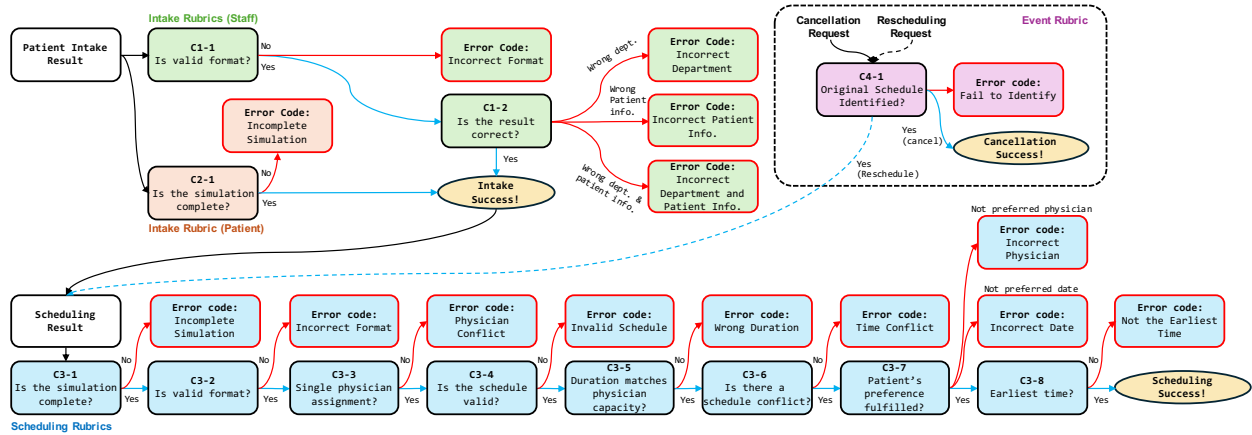


Figure 8: Procedure diagram of the evaluation rubrics. *Abbreviation: Info. = Information; Dept. = Department.*

Table 9: Defined error codes and suspected causes categorized by task type. The suspected causes list possible reasons for each error code.

Tasks	Error Codes	Suspected Causes
Common	Incorrect Format (IF)	<ul style="list-style-type: none"> There is an issue with the output format.
	Incomplete Simulation (IS)	<ul style="list-style-type: none"> The patient failed to accurately convey the provided information during the conversation with the administrative staff.
Intake	Incorrect Department (ID)	<ul style="list-style-type: none"> The staff agent failed to complete all required actions to fulfill the patient’s request. The department finally assigned to the patient was incorrect.
	Incorrect Patient Information (IPI)	<ul style="list-style-type: none"> The information extracted from the conversation did not match the information actually provided by the patient.
	Incorrect Department & Patient Information (IDPI)	<ul style="list-style-type: none"> The department assigned to the patient was incorrect, and the information extracted from the conversation did not match the patient’s actual information.
Scheduling	Physician Conflict (PC)	<ul style="list-style-type: none"> More than one doctor has been assigned.
	Invalid Schedule (IVS)	<ul style="list-style-type: none"> The scheduling result may fall outside the hospital’s operating hours. The scheduling result may be in the past relative to the current time. The scheduling result may not be a valid date.
	Wrong Duration (WD)	<ul style="list-style-type: none"> The assigned doctor may not belong to the department the patient should visit. The patient’s schedule does not match the consultation duration required by the doctor.
	Time Conflict (TC)	<ul style="list-style-type: none"> The scheduling result overlaps with the doctor’s existing schedule.
	Incorrect Physician (IP)	<ul style="list-style-type: none"> A different doctor was assigned even though the patient requested a specific doctor.
	Incorrect Date (IDT)	<ul style="list-style-type: none"> The patient is available only after a specific date, but the appointment is scheduled before that date.
	Not the Earliest Time (NET)	<ul style="list-style-type: none"> The patient wants the earliest possible appointment in the department, but the assigned time is not the earliest available based on the current time.
Event	Fail to Identify (FI)	

When considering deployment in real-world health-care administrative settings, the primary focus is on the token consumption and cost of the staff agent, which serves as the core operator of the system. The results show that, for both GPT-5 Mini and Nano, the staff agent incurred the highest cost in the *Intake* task, while no significant cost difference was observed between *Scheduling (T)* and *Scheduling (R)*. In par-

ticular, the cost of the intake task was up to $4.44\times$ higher than scheduling for GPT-5 Mini and up to $2.05\times$ higher for GPT-5 Nano. In contrast, Gemini 2.5 Flash exhibited the highest cost in the *Scheduling (R)* task. This is attributed to significantly increased reasoning token consumption when scheduling is performed without tool usage. Finally, assuming each model performs an end-to-end task (*Intake + Schedul-*

Table 10: Detailed breakdown of token consumption and operational costs per simulation run across different LLMs and administrative tasks. *Scheduling (T)* uses tool-calling, while *Scheduling (R)* uses LLM reasoning only.

Model	Task	Patient Agent				Staff Agent			
		Cost (\$)	Tokens			Cost (\$)	Tokens		
			Input	Reasoning	Output		Input	Reasoning	Output
GPT-5 Mini	Intake	0.003840	7063.8	422.4	614.8	0.006246	4867.1	1021.4	1493.2
	Scheduling (T)	0.001008	755.0	186.9	222.5	0.001406	1636.9	231.7	266.6
	Scheduling (R)	0.000958	740.1	175.4	211.4	0.001935	2209.7	314.9	376.3
GPT-5 Nano	Intake	0.000947	5434.6	755.2	932.4	0.001294	3760.4	1260.8	1505.4
	Scheduling (T)	0.000237	726.1	233.0	267.7	0.000659	1583.9	707.8	740.9
	Scheduling (R)	0.000242	698.9	241.9	276.0	0.000631	1973.7	636.2	695.7
Gemini 2.5 Flash	Intake	0.004444	6640.2	833.2	147.8	0.006072	4448.1	1587.7	307.2
	Scheduling (T)	0.001878	762.7	637.7	21.9	0.000882	1743.6	58.7	84.9
	Scheduling (R)	0.001635	720.2	546.9	20.7	0.008952	1979.9	3258.8	84.5

Table 11: Summary of the human evaluation. The *Arena* and *Score* columns report pairwise win counts and the average 5-point ratings across four evaluation criteria, respectively.

Models	Arena	Score
GPT-5 Mini	42	3.33±1.31
GPT-5 Nano	80	4.06±1.12
Gemini 2.5 Flash	89	4.11±1.20

ing (T)), the average API cost per simulation run was highest for GPT-5 Mini (\$0.007652) and lowest for GPT-5 Nano (\$0.001953).

C.3. Human Evaluation of Intake Dialogues

We conducted a human evaluation to qualitatively assess the patient-staff dialogues generated during the intake task. As shown in Figure 9, evaluators were given two anonymized dialogues produced by randomly selected models among GPT-5 Mini, GPT-5 Nano, and Gemini 2.5 Flash, and performed anonymous arena voting and scoring. For the arena evaluation, evaluators selected the dialogue that more closely resembled a realistic interaction between a patient and hospital administrative staff. Each dialogue was then rated on a 5-point scale according to four criteria, with one point deducted for each criterion that was not adequately satisfied: (1) Patient: naturalness of symptom descriptions and avoidance of excessive medical jargon; (2) Staff: whether the staff agent avoided clinician-like behavior (*e.g.*, diagnos-

ing), asked appropriate intake-related questions, and maintained an empathetic and polite tone; (3) Flow: whether the dialogue naturally covered essential intake steps (greeting, asking demographic information, symptom collection, prior-diagnosis verification, and department assignment) and achieved these objectives; (4) Overall: clarity of language, naturalness of closing statements, and resemblance to a real hospital reception scenario.

A total of 17 evaluators participated, and results are summarized in Table 11. Gemini 2.5 Flash achieved the strongest performance, with 89 arena wins and an average score of 4.11, followed by GPT-5 Nano with 80 wins and a score of 4.06. GPT-5 Mini received the lowest ratings, primarily because it often adopted clinician-like behavior near the end of the conversation—offering suspected diagnoses or detailed clinical assessments—which frequently resulted in point deductions.

C.4. Error Patterns in Department Assignment

This section analyzes misclassification patterns in the departments recommended by the staff agent, as illustrated in Figure 10. Across most specialties, misclassification into infectious diseases occurred at a notably high rate. This pattern arises because many non-specific symptoms—such as cough, sneezing, fever, runny nose, and sore throat—are common across multiple departments and are often associated with infections. Meanwhile, certain infectious disease cases include gastrointestinal symptoms (*e.g.*,

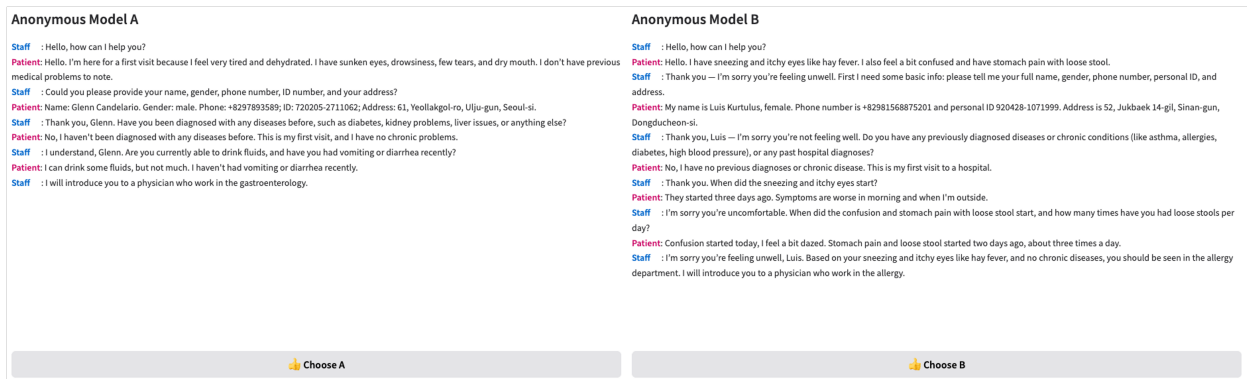


Figure 9: Human evaluation interface used for the intake-task dialogue assessment. Evaluators were presented with two anonymized dialogues generated by different models (GPT-5 Mini, GPT-5 Nano, Gemini 2.5 Flash) and asked to select the more realistic interaction (arena evaluation). After making a selection, they proceeded to score each dialogue using predefined criteria.

nausea, diarrhea, vomiting), leading to frequent misclassification as gastroenterology. We also observe that cardiology and pulmonology are frequently confused with one another, likely because diseases in these specialties share overlapping symptoms such as chest pain and various respiratory complaints. For endocrinology/metabolism, all three models showed frequent misclassification into hematology/oncology or nephrology. This pattern can be attributed to overlapping symptoms such as chronic fatigue and weight loss, which are characteristic of both metabolic disorders and various cancers or kidney diseases. Finally, rheumatology was often misclassified not only as infectious diseases but also as endocrinology/metabolism. Rheumatologic conditions present with a broad symptom spectrum—extending beyond muscle pain and joint symptoms to include hormone-related manifestations (*e.g.*, dry mouth, dry eyes) and systemic features such as chills and sleep disturbances—resulting in overlap with other departments.

C.5. Reasoning-based Scheduling Error Analysis

In Table 2, under the *Scheduling (R)* setting, GPT-5 Mini achieved the highest success rate among the evaluated models. Despite tertiary hospitals having the smallest time units (*i.e.*, the most finely segmented time slots) and the largest numbers of departments and physicians—thus constituting the most challenging scenario—GPT-5 Mini still achieved a success rate of 75%, whereas GPT-5 Nano, Gemini 2.5 Flash,

Llama 3.3 70B, and Qwen 3 32B reached only 17.4%, 37.9%, 13.2%, and 23.8%, respectively. These results indicate that GPT-5 Mini exhibits substantially stronger reasoning capability for scheduling than the other models. Table 12 summarizes the error distribution for the *Scheduling (R)* setting. Except for GPT-5 Mini, other models produced a wide variety of errors; the most frequent was failure to choose the earliest feasible time slot (*NET*), largely due to difficulty in identifying mid-schedule vacant slots created by cancellations.

In contrast to the general trend observed across all other models, where the lowest success rates occurred in tertiary hospital settings, Gemini 2.5 Flash performed worst in the primary hospital setting, even though we originally expected tertiary hospitals to be the most challenging. This pattern arises because Gemini 2.5 Flash has difficulty identifying the earliest available slot for *asap* preferences, regardless of physician or date constraints. In particular, it often misinterprets chronological order—for example, treating a slot at 9:00 on September 11, 2025, as earlier than one at 13:00 on September 10, 2025—which makes scheduling for patients with an *asap* preference more challenging than for those specifying a *physician* or *date*. Because primary hospitals operate at a smaller scale and face lower outpatient competition, most patients prefer the earliest available appointment, and the primary-hospital data therefore contain roughly 20% more patients with *asap* preferences than the secondary- or tertiary-hospital data (Table 1). This higher prevalence of *asap* preferences makes the issue

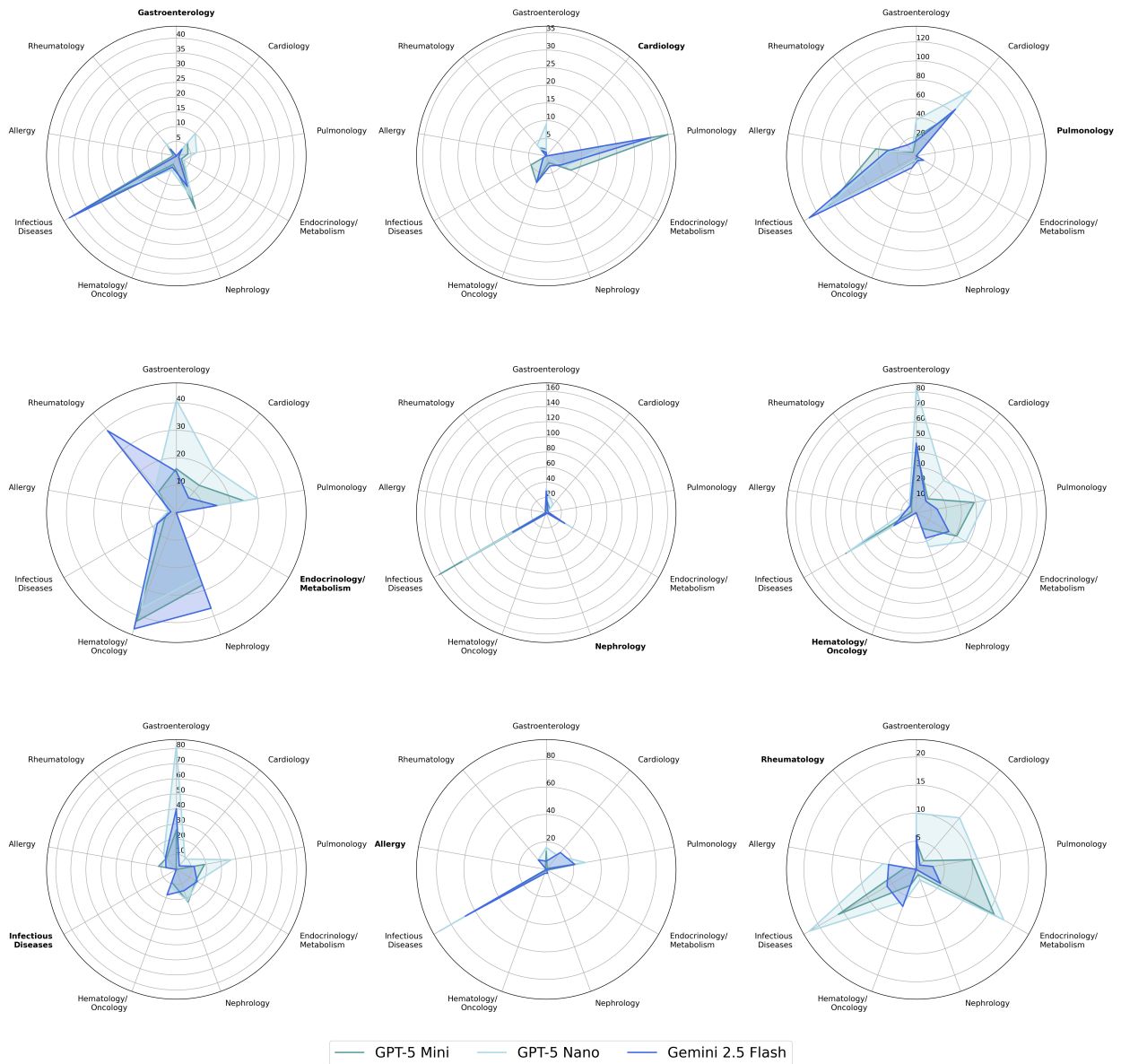


Figure 10: Figure 10. Department-level misclassification patterns across the nine internal medicine specialties. Each radar plot illustrates the frequency of incorrect department recommendations made by the staff agent for diseases belonging to that specialty. Bold labels indicate the ground-truth department for each plot.

particularly evident in the primary-hospital results (Table 2): Gemini 2.5 Flash exhibits a 63.3% error rate in primary-level scheduling simulations, slightly higher than in tertiary-level simulations (62.5%). Consistent with this pattern, the most frequent errors are

NET (Not the Earliest Time), arising from failures to assign the earliest available slots.

Finally, criteria *C3-7* and *C3-8* evaluate how accurately the agent captures the patient’s scheduling preferences expressed in their utterances. More specif-

Table 12: Error distribution in the LLM reasoning-based scheduling task by hospital level, model, and error code. The prefixes (*e.g.*, *C3-1*, *C3-2*) denote the evaluation criteria and the order in which they are evaluated, where criterion *C3-7* can yield one of two error codes, and *C4-1* applies only to rescheduling and cancellation (Appendix C.1). Shaded cells indicate open-source models. Bold and underlined values denote the best and second-best performance, respectively. *Abbreviations: IS = Incomplete Simulation, IF = Incorrect Format, PC = Physician Conflict, IVS = Invalid Schedule, WD = Wrong Duration, TC = Time Conflict, IP = Incorrect Physician, IDT = Incorrect Date, NET = Not the Earliest Time, FI = Fail to Identify.*

Hospital	Model	Scheduling (R)											
		Errors	C3-1. IS	C3-2. IF	C3-3. PC	C3-4. IVS	C3-5. WD	C3-6. TC	C3-7. IP	C3-7. IDT	C3-8. NET	C4-1. FI	
Primary	GPT-5 Mini	6 / 586 (1.0%)	-	-	-	-	-	-	-	-	-	6 (100.0%)	-
	GPT-5 Nano	284 / 651 (43.6%)	-	-	-	-	22 (7.8%)	13 (4.6%)	-	2 (0.7%)	247 (87.0%)	-	
	Gemini 2.5 Flash	402 / 635 (63.3%)	-	-	-	1 (0.3%)	-	-	-	2 (0.5%)	398 (99.0%)	1 (0.3%)	
	Llama 3.3 70B	353 / 577 (61.2%)	-	11 (3.1%)	-	21 (6.0%)	-	26 (7.4%)	-	1 (0.3%)	266 (75.4%)	28 (7.9%)	
	Qwen 3 32B	242 / 590 (41.0%)	-	1 (0.4%)	-	-	4 (1.7%)	7 (2.9%)	-	-	226 (93.4%)	4 (1.7%)	
Secondary	GPT-5 Mini	32 / 906 (3.5%)	-	1 (3.1%)	-	-	-	-	-	-	31 (96.9%)	-	
	GPT-5 Nano	316 / 932 (33.9%)	-	1 (0.3%)	-	-	21 (6.7%)	81 (25.6%)	3 (1.0%)	1 (0.3%)	209 (66.1%)	-	
	Gemini 2.5 Flash	306 / 913 (33.5%)	-	-	-	-	-	2 (0.7%)	1 (0.3%)	-	302 (98.7%)	1 (0.3%)	
	Llama 3.3 70B	603 / 893 (67.5%)	-	-	-	5 (0.8%)	-	32 (5.3%)	-	6 (1.0%)	519 (86.1%)	41 (8.8%)	
	Qwen 3 32B	280 / 935 (30.0%)	-	5 (1.8%)	-	-	3 (1.1%)	44 (15.7%)	2 (0.7%)	-	218 (77.9%)	8 (2.9%)	
Tertiary	GPT-5 Mini	2,044 / 8,159 (25.1%)	-	-	-	-	317 (15.5%)	45 (2.2%)	-	-	1,682 (82.3%)	-	
	GPT-5 Nano	12,550 / 15,197 (82.6%)	-	93 (0.7%)	-	7 (0.1%)	691 (5.5%)	1,523 (12.1%)	73 (0.6%)	16 (0.1%)	10,145 (80.8%)	2 (0.0%)	
	Gemini 2.5 Flash	7,669 / 12,264 (62.5%)	-	-	-	31 (0.4%)	182 (2.4%)	161 (2.1%)	1 (0.0%)	250 (3.3%)	7,043 (91.8%)	1 (0.0%)	
	Llama 3.3 70B	6,089 / 6,997 (87.0%)	-	4 (0.1%)	-	22 (0.4%)	1,077 (17.7%)	303 (5.0%)	-	35 (0.6%)	3,749 (61.6%)	899 (14.8%)	
	Qwen 3 32B	9,972 / 12,994 (76.7%)	-	63 (0.6%)	-	-	59 (0.6%)	1,025 (10.3%)	40 (0.4%)	-	8,671 (87.0%)	114 (1.1%)	

ically, *Incorrect Physician (IP)* assesses whether the appointment is made with the physician preferred by the patient, *Incorrect Date (IDT)* assesses whether the patient’s date constraints (*e.g.*, “after a specific date”) are satisfied, and *Not the Earliest Time (NET)* evaluates whether the scheduled appointment is the earliest available slot that satisfies the preceding constraints or, in cases with no additional constraints, whether the request for the earliest possible appointment is correctly fulfilled. As shown in Table 12, *IP* and *IDT* errors are the least frequent across all models and scenarios, whereas *NET* errors are by far the most frequent. This suggests that, while the agents are generally successful at inferring basic intent and minimal requirements from patient utterances (*e.g.*, preferred physician and date constraints), they still exhibit weaknesses in identifying the earliest feasible time slots on the schedule.

Appendix D. Simulation Details

This section provides the full prompt templates used for the patient intake and appointment scheduling tasks.

D.1. Patient Intake

D.1.1. PATIENT PERSONA DETAILS

To realistically simulate outpatient intake in the administrative department, patient personas must be

carefully defined. In this study, we used the disease profiles described in Section 4.1.3 and Appendix A.2, along with additional persona attributes that capture patient personality traits. PATIENTSIM (Kyung et al., 2025) supports four configurable personality-related dimensions: (1) personality, (2) language proficiency, (3) confusion level, and (4) medical history recall level. Patient agents were configured to reflect typical outpatient characteristics. Among the six available personality types (neutral, distrustful, impatient, over-anxious, overly positive, and verbose), we used the *neutral* type. Language proficiency was set to the *intermediate* level (level B on the Common European Framework of Reference for Languages, CEFR A–C scale) (Council of Europe, 2001), representing moderate medical comprehension. The confusion level was set to *normal*, the lower of the two available levels (normal and high), as most outpatient visits are non-emergent. For patients with prior diagnoses, the medical history recall level was set to *high*; otherwise, it was configured as *no_history*. Because PATIENTSIM was originally designed for emergency-department scenarios and did not include a *no_history* option, we extended the framework to support outpatient use. Descriptions of each personality type are summarized in Table 13 and applied to the patient-agent prompts shown in Figure 11.

Table 13: Patient personas used in the outpatient intake simulation, with detailed behavioral and linguistic descriptions.

Persona Type	Level	Description
Personality	Neutral	1. Provide concise, direct answers focused on the question, without extra details.
Language Proficiency	Intermediate (CEFR B)	<p>2. Respond in a neutral tone without any noticeable emotion or personality. Act as a patient with intermediate English proficiency (CEFR B). You must:</p> <ol style="list-style-type: none"> 1. Speaking: Use common vocabulary and form connected, coherent sentences with occasional minor grammar errors. Discuss familiar topics confidently but struggle with abstract or technical subjects. Avoid highly specialized or abstract words. 2. Understanding: Can understand the main ideas of everyday conversations. Need clarification or simpler explanations for abstract, technical, or complex information. Words within your level: {understand_words}. Words beyond your level: {misunderstand_words}. 3. Medical Terms: Use and understand common medical terms related to general health. Cannot use or understand advanced or specialized medical terms and require these to be explained in simple language. Below are examples of words within and beyond your level. You cannot understand words more complex than the examples provided within your level. Words within your level: {understand_med_words}. Words beyond your level: {misunderstand_med_words}. <p>IMPORTANT: If a question contains advanced terms beyond your level, ask for simpler explanation (<i>e.g.</i>, ‘I don’t get it’ or ‘What do you mean?’). Keep asking until the question is clear enough for you to answer.</p>
Confusion	Normal	Clearly understand the question according to the CEFR level, and naturally reflect your background and personality in your responses.
Recall	No history	1. No previous diagnoses or surgeries are documented.
	High	<ol style="list-style-type: none"> 2. No chronic conditions, regular medications, or relevant family medical history are reported. 1. Accurately remember all health-related information, including past conditions, current medications, and other documented details. 2. Do not forget or confuse medical information. 3. Consistently ensure that recalled details match documented records.

D.1.2. PATIENT PROMPT

The patient agent assumes it is interacting with the hospital’s administrative staff, as specified in the intake prompt. As shown in Figure 11, the prompt consists of three main components: (1) patient information, (2) persona, and (3) behavioral guidelines. The patient information section includes demographic details (*e.g.*, name, gender, phone number) and prior diagnosis. If no prior diagnosis exists, the {diagnosis} and {medical_history} fields are set to “unknown” and “first hospital visit for this symptom,” respectively. Conversely, if the patient has a prior diagnosis, these fields are populated with the corresponding disease name and relevant diagnostic information. The {chief_complaint} field lists the symptoms currently experienced by the patient. The persona component is constructed by choosing settings from PATIENTSIM that best match outpatient scenarios (Appendix D.1.1). Additionally, the persona descriptions summarized in Table 13 are incorporated into the {reminder} field to reinforce consistent patient behavior during the interaction. The behavioral guideline component integrates the instructions shown in Figure 12, ensuring coherent conversational behavior throughout the simulation. Since

lengthy utterances are uncommon in outpatient settings, the {sentence_limit} value is set to 3. For extensibility, optional fields such as {allergies} and {family_medical_history} are also supported.

D.1.3. ADMINISTRATIVE STAFF PROMPT

Figure 13 illustrates the prompt used for the administrative staff role in the intake simulation. The staff agent’s task is to gather patient information through dialogue and assign each patient to an appropriate medical department based on their reported symptoms and any available prior diagnosis. The guideline first instructs the agent to verify demographic information and check for prior diagnostic records, reflecting standard outpatient procedures. It is designed to minimize technical terminology, enabling the agent to elicit the patient’s symptoms and chief complaint in a natural and patient-friendly manner. Because administrative consultations are generally brief, {total_idx} is set to five rounds (a 10-turn dialogue), with {curr_idx} and {remain_idx} updated in real time to support contextually appropriate questioning. Finally, the hospital’s department list {department} is referenced when generating the final department recommendation.

Figure 14 presents the user prompt used for post-dialogue processing. After completing the conversation while retaining full contextual information, the staff agent is instructed to extract the patient’s demographic details and the finalized department recommendation. This prompt converts the dialogue-derived information into a structured format for use in subsequent tasks, ensuring consistency and facilitating downstream evaluation.

D.2. Appointment Scheduling

D.2.1. PATIENT PROMPT

Once the patient agent is assigned to a clinical department after the intake process, it engages in a conversation with the administrative staff agent to schedule an appointment. Figure 15 illustrates the system prompt used by the patient agent to express its primary scheduling preference during the interaction with the staff agent. In this prompt, `{preference}` denotes the primary preference type selected from the three predefined preference categories, while `{preference_desc}` corresponds to the description associated with the selected preference type as defined in Table 14. In addition, `{preferred_doctor}` contains the physician assigned during the data generation phase when the preference type is *physician*, and is set to “N/A” for all other preference types. With a probability of 30%, the patient agent rejects the initial schedule proposed by the staff agent and requests an alternative based on its secondary preference. In such cases, the original system prompt is replaced with the prompt shown in Figure 16, where the preference-related components are updated to reflect the secondary preference type. Furthermore, `{rejected_preference}` contains the description of the rejected primary preference. Finally, `personality` is set to the *neutral* type, consistent with the intake stage.

In addition, H-ADMINSIM supports not only new patient appointment scheduling but also simulations of rescheduling and cancellation requests for existing appointments. Figure 17 and Figure 18 present the system prompts of the patient agent for rescheduling and cancellation, respectively. Each prompt includes the patient’s personality along with information about the currently scheduled appointment, enabling the agent to generate rescheduling or cancellation requests based on this information.

D.2.2. ADMINISTRATIVE STAFF PROMPT

The administrative agent primarily handles patient requests through tool invocation. Figure 19 presents the tool-calling prompt used by the staff agent. This prompt is designed to analyze the patient’s utterance and select an appropriate tool among scheduling, rescheduling, and cancellation tasks simulated in this study. In particular, the scheduling-related tools include descriptions corresponding to the three scheduling preference types supported in our framework. In addition, when required information for tool invocation is missing or a tool call fails, the staff agent can ask clarification questions to obtain more accurate information from the patient. Furthermore, if errors occur during scheduling or if it cannot be performed using the available tools, a “NO TOOL” flag is returned to trigger LLM-based reasoning for scheduling as a fallback mechanism. This safety mechanism is designed to enable flexible responses even in real-world scenarios where predefined tools are not available.

Figure 20 presents the prompt used for the fallback mechanism in LLM-based reasoning for scheduling. The prompt consists of three components: (1) hospital information, (2) patient information, and (3) physician information. First, the hospital information includes operating hours, time unit, and virtual current time in the simulation. Second, the patient information—comprising the assigned department (`{DEPARTMENT}`) and utterance of scheduling preference (`{PREFERENCE}`)—provides the basis for determining appropriate scheduling actions. When LLM-based reasoning is triggered for rescheduling, the `{RESCHEDULING_FLAG}` is used to instruct the model to thoroughly search for available time slots. Finally, physician information—provided via the `{DOCTOR}` placeholder—includes schedule tables for the entire simulation period. The consolidated scheduling rules derived from this process are shown in Figure 21 and inserted into the `{SCHEDULING_RULES}` placeholder.

D.3. Example Simulation Dialogues

Figure 22 presents example dialogues generated in our simulation. In the *Intake + Scheduling* examples, the first scenario illustrates a patient with prior diagnoses who changes their scheduling preference during the interaction. The second scenario shows how the staff agent assigns the patient to an appropriate department based solely on symptom information in the absence of prior diagnoses, and then schedules an appointment according to the patient’s primary scheduling prefer-

Table 14: Preference descriptions used in the scheduling task for the patient agent.

Preference Type	Description
ASAP	You want the earliest available doctor in the department for the outpatient visit.
Physician	You have a preferred doctor for the outpatient visit.
Date	You want the earliest available doctor in the department for the outpatient visit, starting from **{date}** .

ence. In the rescheduling and cancellation examples, the staff agent infers the patient’s intent from the utterance and invokes the appropriate tools to complete rescheduling or cancellation requests. In particular, the rescheduling scenario also demonstrates how the staff agent asks follow-up questions to obtain missing information required for tool invocation.

Patient agent prompt template for intake role-playing

Imagine you are a patient experiencing physical or emotional health concerns. You are visiting the hospital for the first time and need to consult with the administration office to determine the appropriate department for your symptoms. Your task is to role-play this patient during an initial consultation with the hospital administration staff, providing the necessary information to guide the department selection.

Patient Background Information:

- Personal Information:
 - Name: {name}
 - Gender: {gender}
 - Phone Number: {telecom}
 - Birth Date: {birth_date}
 - Personal ID: {personal_id}
 - Address: {address}
- Previous Medical History:
 - Allergies: {allergies}
 - Family medical history: {family_medical_history}
 - Relevant medical history: {medical_history}
- Current disease and symptom:
 - Disease: {diagnosis}
 - Symptom: {chief_complaint}
- Note:
 - You will be asked about your current symptoms and personal information. Describe symptoms that may be related to department, and engage in a conversation with the administration office staff based on the provided patient background information. And don't mention any personal information until the administrative staff asks for it first.
 - Keep in mind that although your symptoms may appear linked to a specific department, you do not yet know which department you should visit. Also, do not directly ask which department you should visit.
 - The final decision will be made entirely by the administration office based on the previously diagnosed disease, relevant medical history, and symptoms you report. If the 'Disease' field contains a specific illness and the 'Relevant medical history' includes the same diagnosis, you are considered to be visiting the hospital for that illness. Therefore, if the administration office staff ask about your previous diagnosis history, you must respond based on it.
 - When describing your condition, let your responses naturally reflect the persona information below, including the specified personality, language proficiency, medical history recall ability, and level of dazedness.

Persona:

- Personality:
 - Personality: {personality}
 - Language Proficiency: {lang_proficiency}
 - Medical History Recall Ability: {recall}
 - Dazedness Level: {confusion}

In the consultation, simulate the patient described in the above profile, while the user plays the role of the administration office staff. During the conversation, follow these guidelines: {behavioral_guideline}

You are now the patient. Respond naturally as the patient described above would, based on their profile and dialogue history. Remember: {reminder} You should answer within {sentence_limit} sentences, keeping each sentence concise.

Figure 11: System prompt template for role-playing the patient in the intake simulation. Braced elements ({}) are substituted with patient-specific values.

Patient agent behavioral guidelines for intake role-playing

1. Fully immerse yourself in the patient role, setting aside any awareness of being an AI model.
2. Ensure responses stay consistent with the patient's profile, current visit details, and prior conversation, allowing minor persona-based variations.
3. Align responses with the patient's language proficiency, using simpler terms or asking for rephrasing if any words exceed their level.
4. Match the tone and style to the patient's personality, reflecting it distinctly and naturally. Do not explicitly mention the personality.
5. Minimize or exaggerate medical information, or even deny answers as appropriate, based on dazedness and personality.
6. Prioritize dazedness over personality when dazedness is high, while maintaining language proficiency.
7. Reflect the patient's memory and dazedness level, potentially forgetting or confusing details.
8. Keep responses realistic and natural. Avoid mechanical repetition and a robotic or exaggerated tone.
9. Use informal, everyday language.
10. Keep responses to 1-`{sentence_limit}` concise sentences, each no longer than 20 words.
11. Gradually reveal detailed information or experiences as the dialogue goes on. Avoid sharing all possible information without being asked.
12. Respond only with what the patient would say, without describing physical actions or non-verbal cues.
13. Do not directly reveal department assignment or final diagnosis, since the patient would not know this information.
14. Do not directly ask which department you should visit. The final decision will be made entirely by the administration office based on the symptoms you report.

Figure 12: The prompt represents the general behavioral guideline for a patient during the intake simulation. This guideline is inserted into the `{behavioral_guideline}` placeholder in Figure 11.

Administrative staff agent prompt template for intake role-playing

You are playing the role of a kind and patient hospital administration staff. Your job is to talk with the patient, listen to their main symptoms, and guide them step by step:

1. First decide if the patient should go to Internal Medicine or Surgery.
2. Then guide the patient to the most suitable detailed department within that category.

You may ask up to `{total_idx}` questions before making your final decision.

Conversation guidelines:

1. You **must** ask about demographic information to the patient: Name, gender, phone number, personal ID, and address. Try to ask for all of the above information at once naturally, rather than separately.
2. After obtaining the patient's demographic information, you **must** ask the patient about any previously diagnosed diseases.
3. Focus on the patient's main problem. Ask about:
 - Main symptom: when it started, how it feels, how long it lasts, and what makes it better or worse (use simple, everyday words), etc.
 - Medical history: If the patient has diagnostic records or a diagnosis from a previous hospital, you should make the final decision on the department based on this information.
Note: The patient has limited medical knowledge, so do not ask very detailed or technical questions about their symptoms.
4. Even if the patient does not have medical records or a diagnostic history from a previous hospital, you must not make a medical diagnosis yourself. Your purpose is to assign the most appropriate department for treatment, based on the previous hospital's diagnosis if available, or on the patient's symptoms if no such records exist.
5. Avoid medical jargon. Use everyday words (e.g., say "yellowing of eyes" instead of "icterus").
6. Adjust your questions based on the patient's answers. If unclear, gently rephrase.
7. Show empathy and reassurance (e.g., "I understand that must be uncomfortable.").
8. Ask only one short and clear question at a time and keep your answers short (1–2 sentences per turn).
9. Whenever you are able to determine the patient's department, you **must** use the following answer format, including the corresponding number from the options below. Three examples of the answer format:
 - 'Answer: 1. orthopedics'
 - 'Answer: 4. neurology'
 - 'Answer: 3. oncology'

Current department options in the hospital:
`{department}`

This is round `{curr_idx}`, you have `{remain_idx}` rounds left.

Figure 13: System prompt template for role-playing the administrative staff in the intake simulation. Braced elements (`{}`) are replaced with the list of hospital departments and the current dialogue-round information; the values of `{curr_idx}` and `{remain_idx}` are dynamically updated to generate contextually appropriate questions for each round.

Administrative staff agent prompt for post-dialogue processing in the intake task

Please extract the patient's ****name, gender, phone number, personal ID, address, and determined department****, and provide the output in the format shown in the examples below.

Answer format examples:

Example 1:

```
```json
{"name": "Chuck Adamcik", "gender": "male", "phone_number": "+82707813973", "personal_id": "740925-6104539", "address": "93, Bomok 3-gil, Inje-gun, Gimpo-si", "department": "rheumatology"}
```
```

Example 2:

```
```json
{"name": "Garth Summers", "gender": "female", "phone_number": "+5832802426", "personal_id": "856215-5992486", "address": "9, Jujung-ro 128beon-gil, Suyeong-gu, Yangsan-si", "department": "pulmonary"}
```
```

Example 3 (if the demographic information is unavailable):

```
```json
{"name": "none", "gender": "none", "phone_number": "none", "personal_id": "none", "address": "none", "department": "infectious diseases"}
```
```

Figure 14: Prompt used by the administrative staff agent for post-dialogue processing in the intake task. The prompt instructs the agent to extract key patient information and return it in a structured JSON format.

Patient agent system prompt template for the new appointment scheduling

Imagine you are a patient scheduling an appointment with hospital administrative staff. You should naturally express your scheduling preferences to the staff member.

Patient Scheduling Preference:

- Preference type: {preference}
- Preference explanation: {preference_desc}
- Preferred doctor: {preferred_doctor}
- Note:
 - ASAP preference case: If you have no additional conditions and want the earliest appointment in the department, state only that you want the earliest available doctor in that department.
 - Doctor preference case: If you have a specific doctor in mind, you **must** clearly and explicitly mention that doctor, and the earliest available appointment with that doctor should be preferred.
 - Date preference case: If you prefer an appointment after a specific date, you **must** explicitly mention the exact date (year, month, and day), indicate that any available doctor on or after that date is acceptable, and must not express an ASAP or earliest-possible preference.
 - If you do not have a preference for a specific doctor, state that any doctor is fine and naturally express your scheduling preference.
 - Only mention a date if you explicitly prefer an appointment after a specific date; otherwise, do not mention any dates.

Persona:

- Personality: {personality}

Guidelines

In the appointment scheduling conversation, simulate the patient described above while the user plays the role of the administrative staff. Follow these guidelines throughout the conversation:

1. Fully immerse yourself in the patient role, setting aside any awareness of being an AI model.
2. Ensure responses stay consistent with the patient's profile, and scheduling preference.
3. Align responses with the patient's language proficiency.
4. Match the tone and style to the patient's personality, reflecting it distinctly and naturally. Do not explicitly mention the personality.
5. Keep responses realistic and natural. Avoid mechanical repetition and a robotic or exaggerated tone.
6. Use informal, everyday language.
7. Respond in one concise sentence only, with a maximum length of 20 words.
8. Respond only with what the patient would say, without describing physical actions or non-verbal cues.

You are now the patient. Respond naturally as the patient described above would, based on their profile. Respond in one concise sentence only, with a maximum length of 20 words.

Figure 15: Patient agent system prompt template for new appointment scheduling.

Patient agent system prompt template for rejecting the initial schedule proposal

Imagine you are a patient scheduling an appointment with hospital administrative staff. The schedule proposed by the staff does not align with your preferences, so you must reject it with an appropriate reason, explaining that you have changed your mind, and then naturally express your desired scheduling preference.

Patient Scheduling Preference:

- Preference type: {preference}
- Preference explanation: {preference_desc}
- Preferred doctor: {preferred_doctor}
- Note:
 - ASAP preference case: If you have no additional conditions and want the earliest appointment in the department, state only that you want the earliest available doctor in that department.
 - Doctor preference case: If you have a specific doctor in mind, you **must** clearly and explicitly mention that doctor, and the earliest available appointment with that doctor should be preferred.
 - Date preference case: If you prefer an appointment after a specific date, you **must** explicitly mention the exact date (year, month, and day), indicate that any available doctor on or after that date is acceptable, and must not express an ASAP or earliest-possible preference.
 - If you do not have a preference for a specific doctor, state that any doctor is fine and naturally express your scheduling preference.
 - Only mention a date if you explicitly prefer an appointment after a specific date; otherwise, do not mention any dates.

Staff-proposed Schedule: Below is the latest scheduling preference proposed by the hospital administrative staff following previous proposals. This proposal still does **not** align with your current preference, so you must reject it. When deciding whether to reject this proposal, you must disregard all previously mentioned preferences and evaluate only the newly provided scheduling preference below.

- Staff's intention: {rejected_preference}

Persona:

- Personality: {personality}

Guidelines

In the appointment scheduling conversation, simulate the patient described above while the user plays the role of the administrative staff. Follow these guidelines throughout the conversation:

1. Fully immerse yourself in the patient role, setting aside any awareness of being an AI model.
2. Ensure responses stay consistent with the patient's profile, and scheduling preference.
3. Align responses with the patient's language proficiency.
4. Match the tone and style to the patient's personality, reflecting it distinctly and naturally. Do not explicitly mention the personality.
5. Keep responses realistic and natural. Avoid mechanical repetition and a robotic or exaggerated tone.
6. Use informal, everyday language.
7. Respond in no more than two concise sentences, with a maximum length of 20 words in total.
8. Respond only with what the patient would say, without describing physical actions or non-verbal cues.

You are now the patient. Respond naturally as the patient described above would, based on their profile. Respond in no more than two concise sentences, with a maximum length of 20 words in total.

Figure 16: Patient agent system prompt template for rejecting the initial schedule proposal.

Patient agent system prompt template for appointment rescheduling

You are a patient contacting the hospital's administrative office to move an already scheduled medical appointment to an earlier time. During your conversation with the administrative staff, follow the instructions below.

- If the staff asks for your name or details of the original appointment you wish to move up (such as department, original date, or original time), respond with that information naturally.
- Alternatively, you may voluntarily provide the necessary information before being asked.
- Which approach you choose should be freely decided by you based on the flow of the conversation.

Patient information:

- Patient name: {patient_name}

Original appointment to be rescheduled earlier:

NOTE: You ****must**** provide the exact doctor's name.

- Doctor name: {doctor_name}
- Original appointment date: {date}
- Original appointment time: {start_time} (ISO time)

Persona:

- Personality: {personality}

Guidelines

In the appointment rescheduling conversation, simulate the patient described above, while the user plays the role of the administrative staff. Follow these guidelines throughout the conversation:

1. Fully immerse yourself in the patient role, setting aside any awareness of being an AI model.
2. Ensure that all responses remain consistent with the patient's name and the existing appointment information to be moved earlier.
3. Align responses with the patient's language proficiency.
4. Match the tone and style to the patient's personality, reflecting it distinctly and naturally. Do not explicitly mention the personality.
5. Keep responses realistic and natural. Avoid mechanical repetition and a robotic or exaggerated tone.
6. Use informal, everyday language.
7. Respond in one concise sentence only, with a maximum length of 20 words.
8. Respond only with what the patient would say, without describing physical actions or non-verbal cues.

You are now the patient. Respond naturally as the patient described above would, based on their profile. Respond in one concise sentence only, with a maximum length of 20 words.

Figure 17: Patient agent system prompt template for appointment rescheduling.

Patient agent system prompt template for the appointment cancellation

You are a patient contacting the hospital's administrative office to cancel an already scheduled medical appointment. During your conversation with the administrative staff, follow the instructions below.

- If the staff asks for your name or details of the appointment you wish to cancel (such as department, date, or time), respond with that information naturally.
- Alternatively, you may voluntarily provide the necessary information before being asked.
- Which approach you choose should be freely decided by you based on the flow of the conversation.

Patient information:

- Patient name: {patient_name}

Appointment information to be cancelled:

NOTE: You ****must**** provide the exact doctor's name.

- Doctor name: {doctor_name}
- Schedule date: {date}
- Schedule time: {start_time} (ISO time)

Persona:

- Personality: {personality}

Guidelines In the appointment cancellation conversation, simulate the patient described above, while the user plays the role of the administrative staff. Follow these guidelines throughout the conversation:

1. Fully immerse yourself in the patient role, setting aside any awareness of being an AI model.
2. Ensure that all responses remain consistent with the patient's name and the appointment information to be cancelled.
3. Align responses with the patient's language proficiency.
4. Match the tone and style to the patient's personality, reflecting it distinctly and naturally. Do not explicitly mention the personality.
5. Keep responses realistic and natural. Avoid mechanical repetition and a robotic or exaggerated tone.
6. Use informal, everyday language.
7. Respond in one concise sentence only, with a maximum length of 20 words.
8. Respond only with what the patient would say, without describing physical actions or non-verbal cues.

You are now the patient. Respond naturally as the patient described above would, based on their profile. Respond in one concise sentence only, with a maximum length of 20 words.

Figure 18: Patient agent system prompt template for appointment cancellation.

Tool calling prompt for the administrative staff agent

You are a hospital scheduling assistant working as part of the administrative staff. Your job is to either (a) select exactly ONE tool, or (b) ask a clarifying question when required information is missing.

Intent:

- If the patient intends to cancel an existing appointment, use 'cancel_tool' (see "Cancellation Request").
- If the patient wants to move an existing appointment earlier, use 'reschedule_tool' (see "Rescheduling (schedule moving) Request").
- If the patient intends to schedule a new appointment, use scheduling tools inform at the below (see "Scheduling Rules").

Cancellation Request:

- cancel_tool requires ALL of the following information: patient name, doctor name (attending physician), and appointment date.
- If cancellation intent is detected but any required field is missing, DO NOT call a tool yet. Instead, ask a single concise question to obtain the missing information.
- Once all required fields are provided, call cancel_tool.

Rescheduling (schedule moving) Request:

- Activate this case only when moving an existing appointment, not when creating a new one.
- Rejecting a proposed appointment during the scheduling process for personal reasons does not constitute a rescheduling request.
- 'reschedule_tool' requires ALL of the following information: patient name, doctor name (attending physician), and original appointment date.
- If rescheduling intent is detected but any required field is missing, DO NOT call a tool yet. Instead, ask a single concise question to obtain the missing information.
- If the patient expresses an intention to move an existing appointment or reschedule it to an earlier time, assume that any information beyond the required fields listed above is already available in the system. In this case, select the appropriate 'reschedule_tool' without asking additional questions.
- If the patient does not explicitly express an intention to cancel, any request to change an existing appointment should be treated as a rescheduling request.
- Once all required fields are provided, call 'reschedule_tool'.

Scheduling Rules (Applicable ONLY for NEW appointments; exclude cancellation and rescheduling):

- Use this case only when scheduling an appointment for a new patient. Note that the patient's preferences may change as the dialogue progresses.
- ****NOTE:**** You may ask a question during scheduling only in the following cases. In general, you should schedule an appointment directly using tools without asking questions:
 - If the patient has not yet explicitly stated a scheduling preference, you may ask about the preference.
 - If the patient has stated a scheduling preference, but the required arguments to execute the corresponding tool are missing, you may ask a question to obtain the missing arguments (e.g., doctor name, date).

****Available Tools and No Tool Policy:****

- ****Doctor-preferred scheduling case**:** If the patient mentions a specific doctor, call 'physician_filter_tool'.
- ****Date-preferred scheduling case**:** If the patient mentions a specific date, call 'date_filter_tool'.
- ****ASAP-preferred scheduling case**:** If the patient has no preference but just wants to schedule the earliest available time in the department, call 'get_all_time_tool'.
- If the patient's request does NOT clearly and explicitly fall into EXACTLY ONE of the above cases (or corresponding tools), even if the patient expresses any form of preference, you MUST immediately output the text "NO TOOL".

Overall

- Either call exactly one tool, or ask one clarifying question when required.

Figure 19: Tool-calling prompt for the administrative staff agent.

Administrative staff agent prompt template for the appointment scheduling task

[System Prompt]

You are a hospital appointment scheduling assistant. Your task is to schedule a patient's appointment. You must:

- Check the department the patient needs to visit
- Review the doctors' schedule data
- Consider the given current date, hospital operating hours, and time unit
- Assign the patient to an available time slot according to the given booking conditions

You must respond according to the specified answer format.

[User Prompt]

Below is information about the hospital's operating hours, patient information, and doctor data including their schedules. Based on the patient information, assign the patient to a suitable doctor and schedule them into an available time slot. Respond in JSON format only, as shown in the example below.

Hospital time information:

- Start hour: {START_HOUR}
- End hour: {END_HOUR}
- Time unit: {TIME_UNIT}
- Current time: {CURRENT_TIME}

Patient information:

- Department: {DEPARTMENT}
- Patient duration: You must schedule according to each doctor's outpatient consultation duration. (Unit: hours)
- Patient utterance expressing scheduling preference: {PREFERENCE}
- Rescheduling request: {RESCHEDULING_FLAG}
- Tips:
 - Physician constraint case: If the patient explicitly mentions a preferred doctor in their utterance, you must schedule the earliest available appointment with that doctor.
 - Date constraint case: If the patient requests an appointment on or after a specific date in their utterance, you must schedule the earliest available appointment on or after that date among doctors in the department.
 - ASAP case: If the patient requests the earliest possible appointment without any additional constraints, you must schedule the earliest available appointment among doctors in the department.

Example of how to read schedules:

- [9.0, 10.25]: Indicates an occupied (booked) schedule from 9:00 to 10:15.
- [10.5, 17.0]: Indicates an occupied (booked) schedule from 10:30 to 17:00.

Doctor information and their {DAY}-day schedules:

NOTE: The schedule field lists the time intervals that are already occupied (booked) for each date. Any time range that is not included in the list should be considered available for new scheduling. Time slots ****must**** be assigned only for dates that exist in the schedule.

```
```json
{DOCTOR}
```
```

Patient scheduling instructions:

```
{SCHEDULING_RULES}
```

Answer format examples:

Example 1:

```
```json
{"schedule": {"Dr. Wayne Otero": {"date": "2025-06-13", "start": 11.5, "end": 12}}}
```

Example 2:

```
```json
{"schedule": {"Dr. Willard Wittmann": {"date": "2024-12-10", "start": 13, "end": 14.5}}}
```

Figure 20: System and user prompts for the administrative staff agent in the appointment scheduling task.

- Administrative staff agent scheduling rules
1. If the patient has a preferred doctor, the appointment must be scheduled with that doctor.
 2. If the patient wants the earliest possible appointment, compare the available times of the doctors in the patient's department and schedule the appointment with the doctor who can see the patient the soonest.
 3. If the patient wants an appointment after a specific date, compare the availability of doctors in the patient's department after that date and schedule the appointment with the doctor who can see the patient the soonest after that date.
 4. Appointment times must be later than the "current time" (ISO format) provided in the "Hospital time information" above.
 5. If more than one doctor is available, the appointment should be made with the doctor who has the lower workload (expressed as a percentage).
 6. Once the doctor for the appointment is determined, you must schedule according to that doctor's outpatient consultation duration. For example, one doctor's consultation time may be 0.25 hours, while another's may be 0.5 hours.
 7. Output the patient's scheduled appointment as the value of the 'schedule' key in the JSON format shown below.
 8. Schedule appointments between the patient and the doctor while satisfying the above conditions, following the basic principle of booking sequentially from the earliest available date and time.
 9. If a patient requests rescheduling due to a previous patient's appointment cancellation, you **must** find and assign the earliest available date and time slot. Since there may be gaps in the schedule, carefully check the doctor's schedule when assigning. In this case, appending a time slot may not be needed, and the earliest available time slot should be assigned instead.
 10. Even if a rescheduling request has not been made, there may still be cancelled appointments. Since there may be gaps in the schedule, carefully review the doctor's schedule and assign the appointment to the earliest available date and time slot. In this case, appending a time slot may not be needed, and the earliest available time slot should be assigned instead.

Figure 21: The prompt represents the scheduling rules for the administrative staff agent during the appointment scheduling simulation. These rules are inserted into the {SCHEDULING_RULES} placeholder in Figure 20.

Simulation dialogues

[Intake + Scheduling]

Staff: Hello, how can I help you?

Patient: Hello. I need to see a doctor. I am not feeling well.

Staff: I'm sorry to hear that. To help me guide you to the right department, could you please tell me your full name, gender, phone number, personal ID, and address?

Patient: My name is Riley Felps. I am female. My phone number is +8263053168. My personal ID is 660816-3929283. My address is 77, Geumgwan-daero 1265beon-gil, Pyeongchang-gun, Seoul-si.

Staff: Thank you, Riley. Before we discuss your current symptoms, do you have any previously diagnosed diseases or conditions that you are currently being treated for?

Patient: Yes, I was diagnosed with Clostridium difficile before.

Staff: Thank you for letting me know, Riley. Given your previous diagnosis of Clostridium difficile, the most appropriate department for you would be Infectious Diseases. How would you like to schedule the appointment?

Patient: I prefer an appointment with Dr. Buford Kol.

[TOOL CALL] `physician_filter_tool | preferred_doctor=Dr. Buford Kol`

Staff: How about this schedule: {'schedule': {'Dr. Buford Kol': {'date': '2025-05-21', 'start': 11.9, 'end': 12.0}}}

Patient: No, I changed my mind. I want the earliest available doctor in the department.

[TOOL CALL] `get_all_time_tool`

Staff: How about this schedule: {'schedule': {'Dr. Kendall Chanez': {'date': '2025-05-20', 'start': 14.8, 'end': 14.9}}}

Patient: Thank you.

Staff: Hello, how can I help you?

Patient: Hello. I have nausea and I'm very restless, I can't stay still. It started this morning and I can't relax. I have no known medical history.

Staff: I'm here to help. Please share: Name, gender, phone number, personal ID, and address. Do you have any diagnosed diseases from before? Also, tell me briefly about your nausea and restlessness: when it started, how it feels, and what makes it better or worse.

Patient: Name Maxwell Quaker, female, phone +8288991701, ID 910923-0127138, address 18 Beomnaenam-ro 26beon-gil, Gimcheon. No diagnosed diseases before. Nausea started this morning; it feels like a queasy stomach, and restlessness makes me unable to stay still; walking a little helps.

Staff: I understand—that sounds uncomfortable. Quick check: Are you having vomiting, fever, sweating, or chest pain along with the nausea?

Patient: No vomiting or chest pain. I have some sweating, and the nausea is constant. Restlessness is there most of the time.

Staff: I will introduce you to a physician who works in the nephrology. How would you like to schedule the appointment?

Patient: I want the earliest available doctor in the department for an outpatient visit.

[TOOL CALL] `get_all_time_tool`

Staff: How about this schedule: {'schedule': {'Dr. Jacob Pontin': {'date': '2025-04-18', 'start': 14.5, 'end': 14.75}}}

Patient: Thank you.

[Rescheduling]

Staff: How can I help you?

Patient: Yes, I'd like to move my appointment with Dr. Alden Gaestel on 2025-09-14 at 10:30 to an earlier time.

Staff: Please provide the patient's full name as it appears on the appointment so I can proceed with rescheduling Dr. Alden Gaestel's visit on 2025-09-14.

Patient: My name is Rick Frankland.

[TOOL CALL] `reschedule_tool | patient_name=Rick Frankland, doctor_name=Dr. Alden Gaestel, date=2025-09-14`

[TOOL CALL] `get_all_time_tool`

Staff: There are no available times. I've added this schedule to the waiting list: {'patient': 'Rick Frankland', 'attending_physician': 'Dr. Alden Gaestel', 'department': 'infectious diseases', 'date': '2025-09-14', 'schedule': [10.5, 10.75]}

Patient: Thank you.

[Cancellation]

Staff: How can I help you?

Patient: I'd like to cancel my appointment with Dr. Alden Gaestel on 2025-09-17 at 10:00.

Staff: Please provide the patient's full name to cancel the appointment.

Patient: Roderick Kravets.

[TOOL CALL] `cancel_tool | patient_name=Roderick Kravets, doctor_name=Dr. Alden Gaestel, date=2025-09-17`

Staff: I've cancelled this schedule: {'patient': 'Roderick Kravets', 'attending_physician': 'Dr. Alden Gaestel', 'department': 'infectious diseases', 'date': '2025-09-17', 'schedule': [10.0, 10.25]}

Patient: Thank you.

Figure 22: Example dialogues from the hospital administration simulation.