

**CS4514-Project | 22CS170**

# **Web-based elderly home management system**

**Chan Tsz Ngai, Matthew (56213172)**





# Agenda

01 *Introduction*

---

02 *System design*

---

03 *UML diagrams showcase*

---

04 *Methodology and implementation*

---

05 *Algorithm design*

---

06 *Result and evaluation*

---

07 *Conclusion*

---

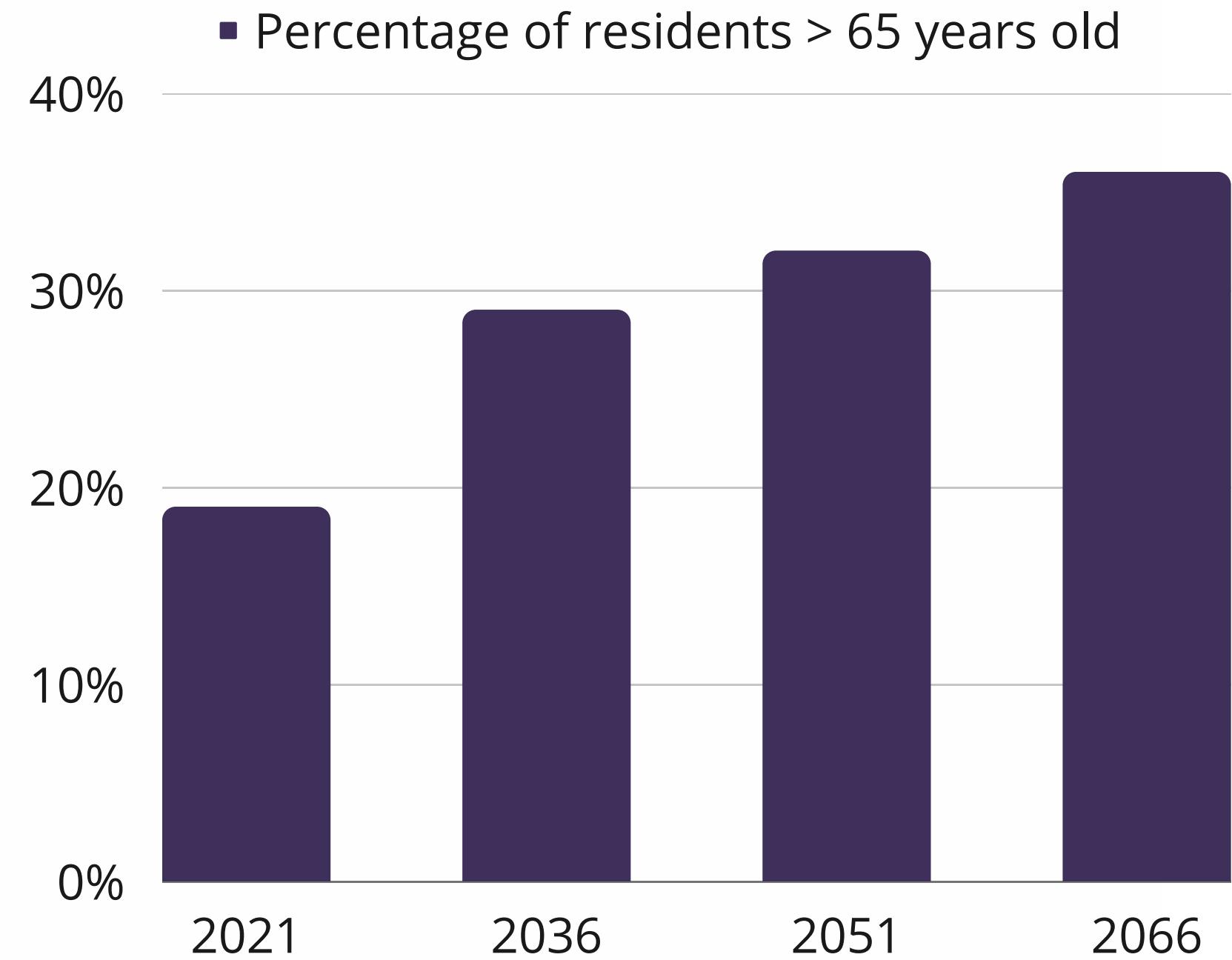
01

# Introduction



# Aging problems

Almost **1/3** of Hong Kong citizens is **aged over 65** in 2066



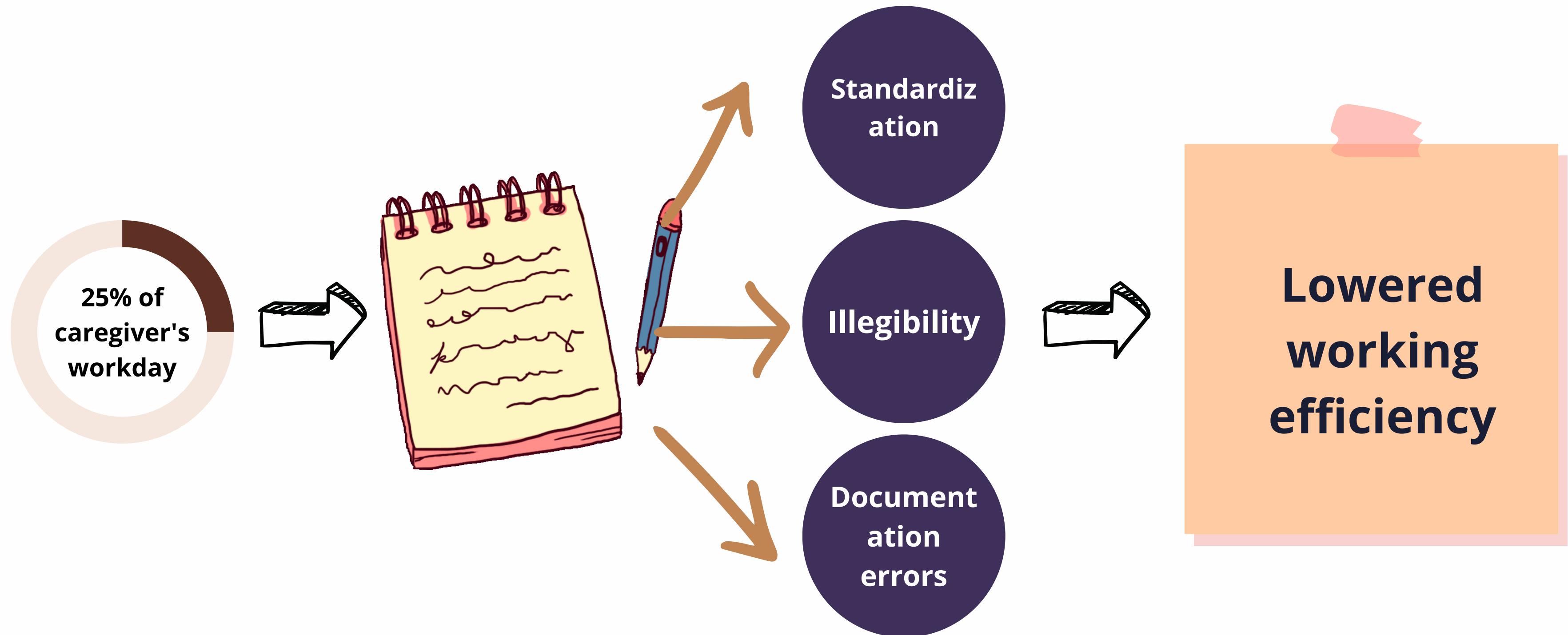
# Lacking resources on elderly home

- lack of incentives  
for private developers

- long processing time  
for approval for RCHE

**Pressure on  
elderly homes  
and caregivers**

# Traditional documentation: Documentation Issues



# Management system: Text-orientated

<b>第一部份：長者資料</b>			
長者姓名：	PT Test9 (中)	PT Test9 (英)	
會員編號：	CT0009	社區券證明書編號：	CCSV09
申請編號：	SPA-CT0009-1912	提供服務時段：	2019 年 12 月
<b>第二部份：服務資料 (請在適當的方格 = 加上 "✓")</b>			
社區券價值：	<input type="radio"/> \$4020 <input type="radio"/> \$5810 <input type="radio"/> \$7260 <input type="radio"/> \$8150 <input checked="" type="radio"/> \$9600		
共同付款級別：	<input checked="" type="radio"/> I (綜合社會保障援助688) <input type="radio"/> II <input type="radio"/> III <input type="radio"/> IV <input type="radio"/> V <input type="radio"/> VI		
應付共同付款金額：	HK\$ 488		
<b>第三部份：每月使用服務</b>			
服務項目	服務數量		
<input type="checkbox"/> 全時間日間中心服務(每週6天)	(每週6天)		
<input checked="" type="checkbox"/> 日間照顧服務(4小時)	10.00 (個數)		
<input type="checkbox"/> 在正常服務時間以外的日間照顧服務	(小時)		
<input checked="" type="checkbox"/> 車輛接送服務(往返中心)	12.00 (次數)		
<input type="checkbox"/> 踏同接送服務(往返住所與車輛)	(次數)		
<input checked="" type="checkbox"/> 到戶護理服務 (專業人員提供)	3.00 (小時)		
<input checked="" type="checkbox"/> 到戶護理服務 (輔助人員提供)	16.00 (小時)		
<input type="checkbox"/> 家居服務	(小時)		
<input type="checkbox"/> 運送服務	(小時)		
<input type="checkbox"/> 送飯服務	(餐數)		
<input type="checkbox"/> 住宿暫託服務	(日數)		
<input type="checkbox"/> 日間到戶看顧 (專業人員提供)	(小時)		
<input type="checkbox"/> 日間到戶看顧 (輔助人員提供)	(小時)		
<input type="checkbox"/> 到戶照顧培訓照顧者 (專業人員提供)	(小時)		
<input type="checkbox"/> 到戶照顧培訓照顧者 (輔助人員提供)	(小時)		
<input type="checkbox"/> 由專業人員提供家居環境安全評估及改善建議	(小時)		
<input type="checkbox"/> 服務時間以外的到戶照顧	(小時)		
<input type="checkbox"/> 到戶復康運動 (專業人員提供)	(小時)		
<input type="checkbox"/> 到戶復康運動 (輔助人員提供)	(小時)		
<input type="checkbox"/> 由輔助人員提供的個人照顧服務	(小時)		
<input type="checkbox"/> test1234	(小時)		
<input type="checkbox"/> test20191025	(小時)		
<input checked="" type="checkbox"/> 服務時間以外的到戶照顧：			
<input checked="" type="checkbox"/> 到戶護理服務 (輔助人員提供)	4.00 (小時)		
<b>第四部份：服務確認及簽署</b>			
認可服務單位負責人簽署：	社區券持有人/家人/照顧者"簽署：		
認可服務單位負責人姓名：	Chan Tai Man		
認可服務單位負責人職位：	CCSV協議簽名人職位	姓名：	
日期：	2019 年 12 月 03 日	簽發收據日期：	

月份: 2009/4	上機用戶: RICKY - 2019年3月14日 11:40:02	終止用戶:																																																																																								
帳號:	登入用戶: RICKY - 2019年3月20日 20:46:19																																																																																									
<table border="1"> <tr> <td>院友編號: 00001</td> <td>床位編號: 01</td> <td>房間類型: 單人房</td> <td>組別: A區</td> </tr> <tr> <td>姓名(中文): 陳大文</td> <td>姓名(英文): CHAN TAI MAN</td> <td colspan="2"></td> </tr> <tr> <td>身份證號碼: A123456(7)</td> <td>出生日期: 1920年3月4日</td> <td>年齡: 99</td> <td>性別: 男</td> </tr> <tr> <td>郵文電碼:</td> <td>簽發日期:</td> <td colspan="2">已申請身份證括免證明書: 否</td> </tr> <tr> <td>電話(1):</td> <td>入住前往址:</td> <td colspan="2"></td> </tr> <tr> <td>電話(2):</td> <td></td> <td colspan="2"></td> </tr> <tr> <td>電話(3):</td> <td>院友分類: 寢位</td> <td>不可以外出</td> <td>護理種類: 全護理</td> </tr> <tr> <td>收訂金日期:</td> <td>訂金:</td> <td>經濟狀況: 綜合援助</td> <td>保障部資料:</td> </tr> <tr> <td>按金應收:</td> <td>按金已收: 0</td> <td>統援號碼: SSP-123456</td> <td>2019年12月31日</td> </tr> <tr> <td>費用(1) 費用:</td> <td>10,000</td> <td>費用(3) 舉片:</td> <td>1,920</td> <td>費用(5) 電視費(月費):</td> <td>150</td> </tr> <tr> <td>費用(2) 特別費用:</td> <td>2,000</td> <td>費用(4) 電動床+電器:</td> <td>300</td> <td>費用(6):</td> <td>&gt;</td> </tr> <tr> <td>入住日期: 2018年1月1日</td> <td>起租日期: 2018年1月1日</td> <td>下期收款: 2019年3月1日</td> <td>結算:</td> <td>預繳設定</td> </tr> <tr> <td>離院日期:</td> <td>離院種類:</td> <td>離院原因:</td> <td>否</td> <td>分單設定</td> </tr> <tr> <td colspan="6">         緊急情況時之聯絡人: 1          聯絡人姓名: 陳小姐          身份證號碼:          與院友關係: 母女          電話(1): 9876 5432          電話(2):       </td> </tr> <tr> <td colspan="6">         藥物敏感: 不詳       </td> </tr> <tr> <td colspan="6">         已離院-未結算-總人數: (0)       </td> </tr> <tr> <td colspan="6">         未離院(全部): 男(1) 女(0) 徒製(1)       </td> </tr> <tr> <td colspan="6">         綜合援助-30日內到期-總人數: (0)       </td> </tr> </table>			院友編號: 00001	床位編號: 01	房間類型: 單人房	組別: A區	姓名(中文): 陳大文	姓名(英文): CHAN TAI MAN			身份證號碼: A123456(7)	出生日期: 1920年3月4日	年齡: 99	性別: 男	郵文電碼:	簽發日期:	已申請身份證括免證明書: 否		電話(1):	入住前往址:			電話(2):				電話(3):	院友分類: 寢位	不可以外出	護理種類: 全護理	收訂金日期:	訂金:	經濟狀況: 綜合援助	保障部資料:	按金應收:	按金已收: 0	統援號碼: SSP-123456	2019年12月31日	費用(1) 費用:	10,000	費用(3) 舉片:	1,920	費用(5) 電視費(月費):	150	費用(2) 特別費用:	2,000	費用(4) 電動床+電器:	300	費用(6):	>	入住日期: 2018年1月1日	起租日期: 2018年1月1日	下期收款: 2019年3月1日	結算:	預繳設定	離院日期:	離院種類:	離院原因:	否	分單設定	緊急情況時之聯絡人: 1 聯絡人姓名: 陳小姐 身份證號碼: 與院友關係: 母女 電話(1): 9876 5432 電話(2):						藥物敏感: 不詳						已離院-未結算-總人數: (0)						未離院(全部): 男(1) 女(0) 徒製(1)						綜合援助-30日內到期-總人數: (0)					
院友編號: 00001	床位編號: 01	房間類型: 單人房	組別: A區																																																																																							
姓名(中文): 陳大文	姓名(英文): CHAN TAI MAN																																																																																									
身份證號碼: A123456(7)	出生日期: 1920年3月4日	年齡: 99	性別: 男																																																																																							
郵文電碼:	簽發日期:	已申請身份證括免證明書: 否																																																																																								
電話(1):	入住前往址:																																																																																									
電話(2):																																																																																										
電話(3):	院友分類: 寢位	不可以外出	護理種類: 全護理																																																																																							
收訂金日期:	訂金:	經濟狀況: 綜合援助	保障部資料:																																																																																							
按金應收:	按金已收: 0	統援號碼: SSP-123456	2019年12月31日																																																																																							
費用(1) 費用:	10,000	費用(3) 舉片:	1,920	費用(5) 電視費(月費):	150																																																																																					
費用(2) 特別費用:	2,000	費用(4) 電動床+電器:	300	費用(6):	>																																																																																					
入住日期: 2018年1月1日	起租日期: 2018年1月1日	下期收款: 2019年3月1日	結算:	預繳設定																																																																																						
離院日期:	離院種類:	離院原因:	否	分單設定																																																																																						
緊急情況時之聯絡人: 1 聯絡人姓名: 陳小姐 身份證號碼: 與院友關係: 母女 電話(1): 9876 5432 電話(2):																																																																																										
藥物敏感: 不詳																																																																																										
已離院-未結算-總人數: (0)																																																																																										
未離院(全部): 男(1) 女(0) 徒製(1)																																																																																										
綜合援助-30日內到期-總人數: (0)																																																																																										
院友項目 院友文件 行政 社工 護理 因入院 留免2018年1月電動床收費																																																																																										
病歷記錄 大便失禁, 中風, 冠心病																																																																																										
藥物敏感 不詳																																																																																										
新增 更改 刪除 儲存 列印 Label  <input type="button" value="搜尋"/>  全勝 未離院 <- <- > >  <input type="button" value="消除"/>   <input type="button" value="退出"/>																																																																																										



## High demand on elderly homes

- ageing population
- lacking resources impose higher pressure on both elderly homes and caregivers

## Low working efficiency inside elderly homes

- long documentation process
- ineffective and error-prone manual written records
- management system quality issues

# Project Objectives



**TRANSFORM THE PAPER-BASED MANAGEMENT IN ELDERLY HOMES**



**REDUCE POTENTIAL DOCUMENTATION ERRORS**



**OPTIMIZE WORKFLOW EFFICIENCY AND CARING SERVICES PERFORMANCE**



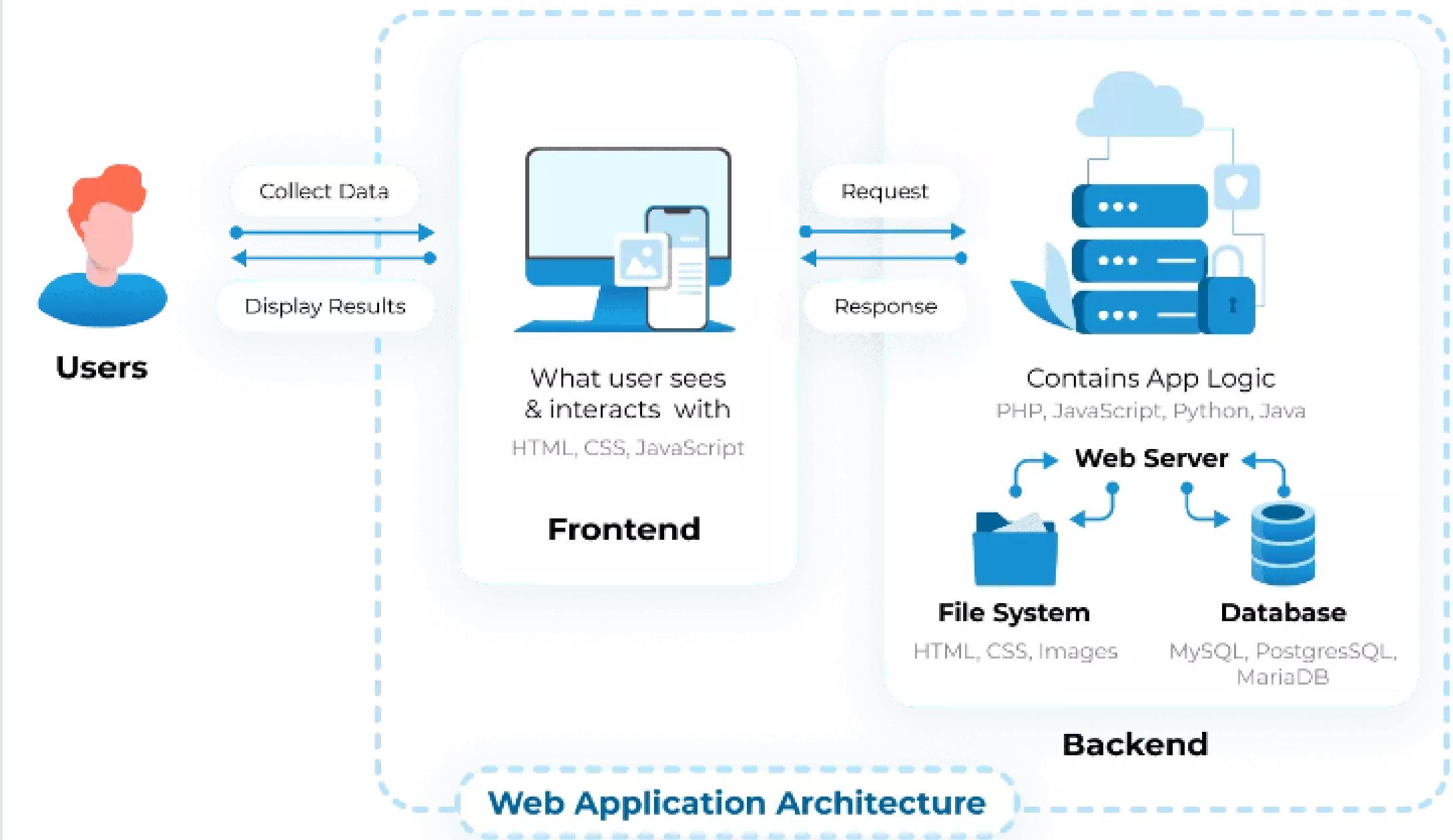
02

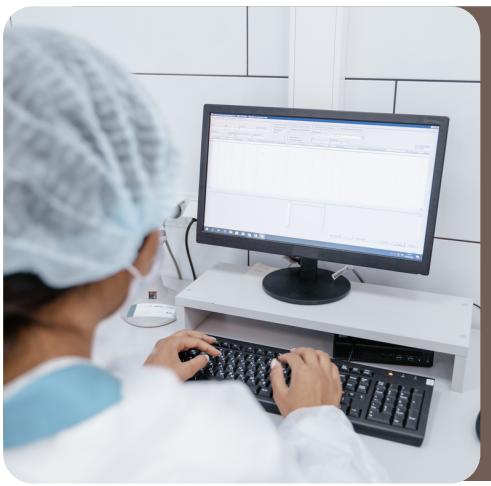
# System design

1

# Three-tier architecture

*The best way for data storage and data retrieval*





## Administrative colleagues

*Management*



## Frontline caregivers

*Care services*



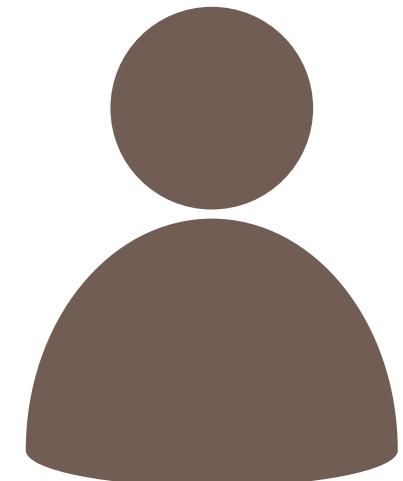
## Residents' relatives

*Status checking*

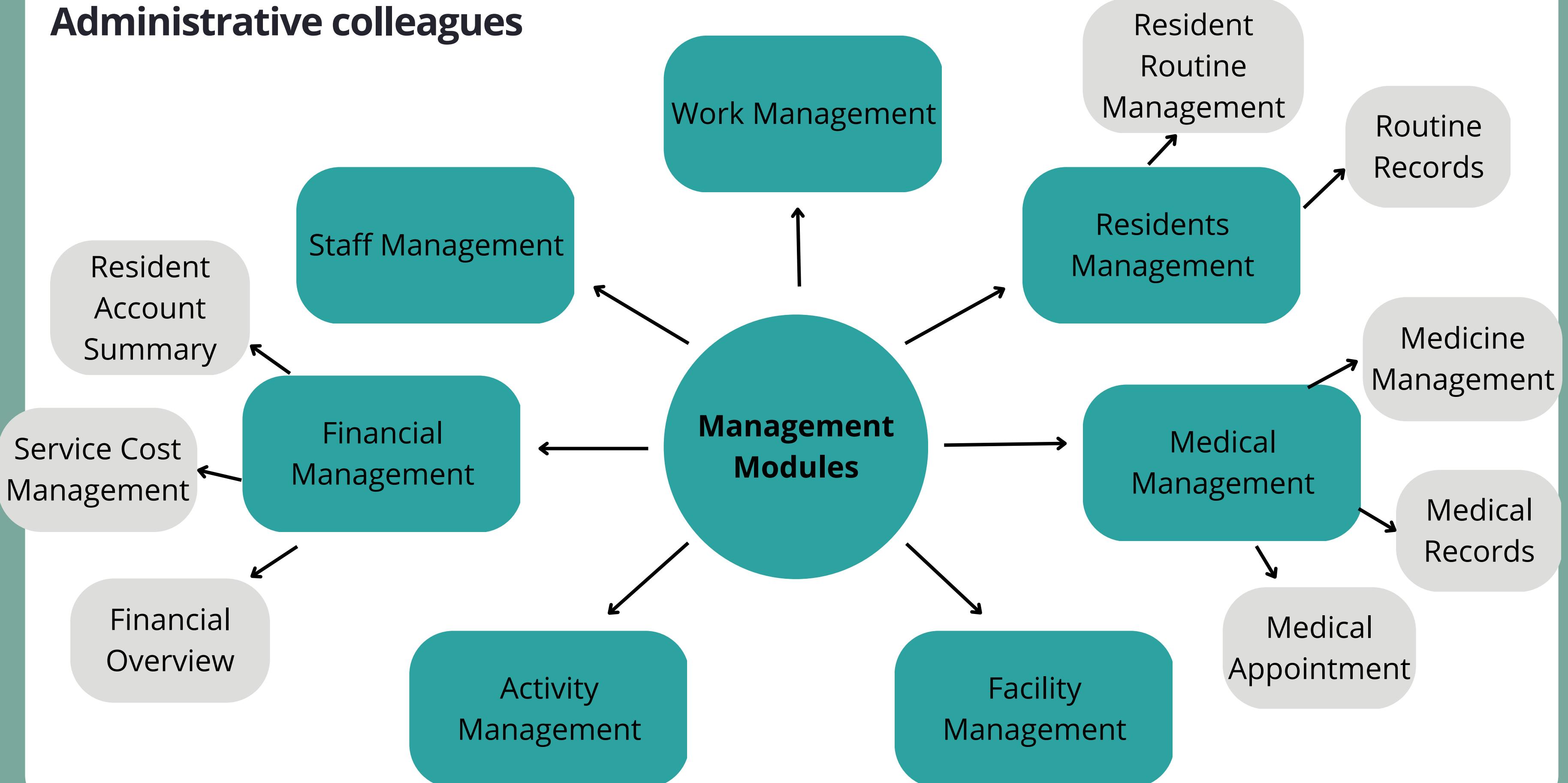
2

# Three separate user interfaces

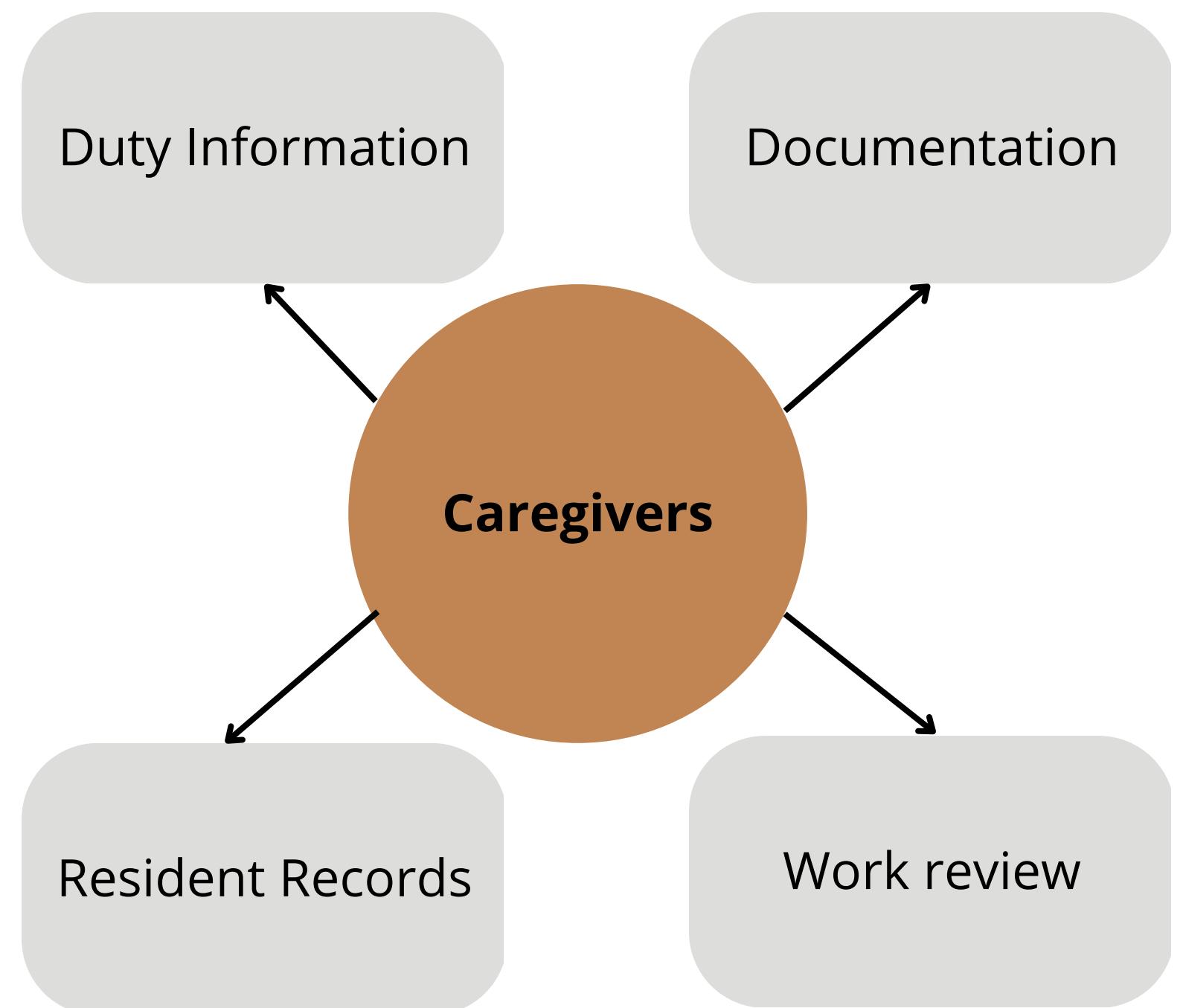
*Different user interfaces based on users' privilege*



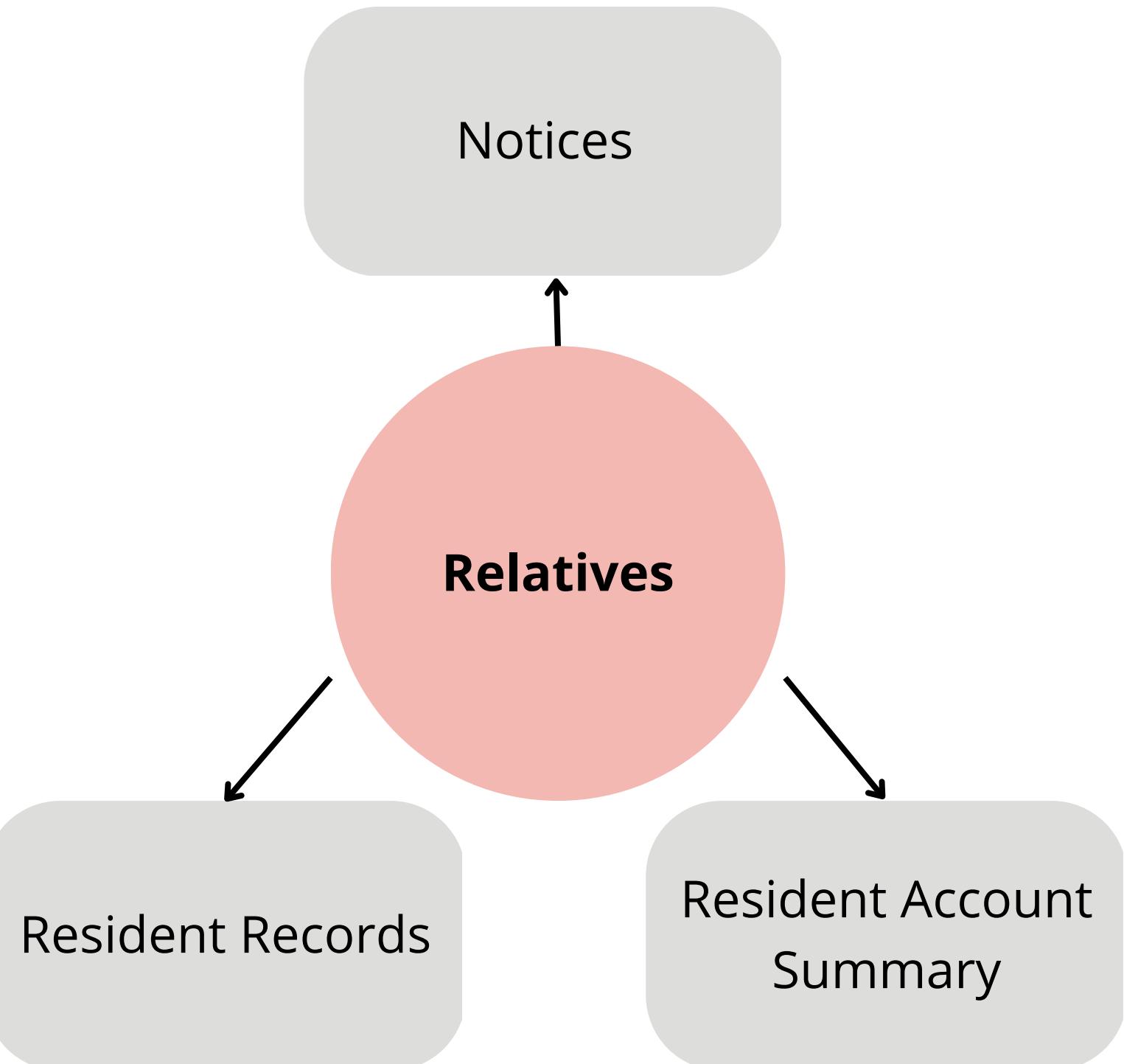
# Administrative colleagues



## Caregivers



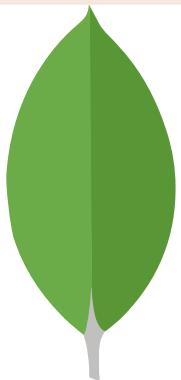
## Relatives



3

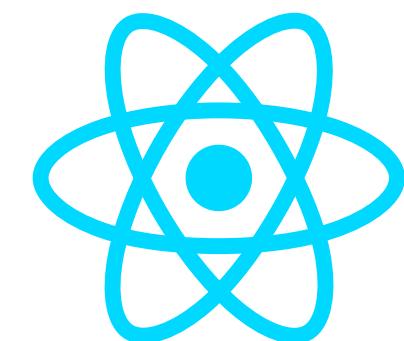
# Four web technologies implementation

*Modern and intertwined web technologies  
comprised of Javascript*

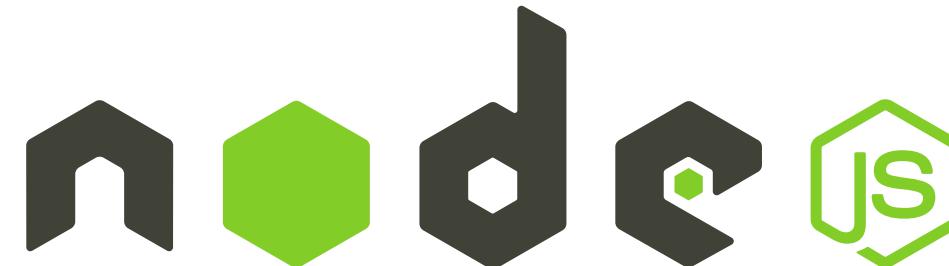


mongoDB

Express

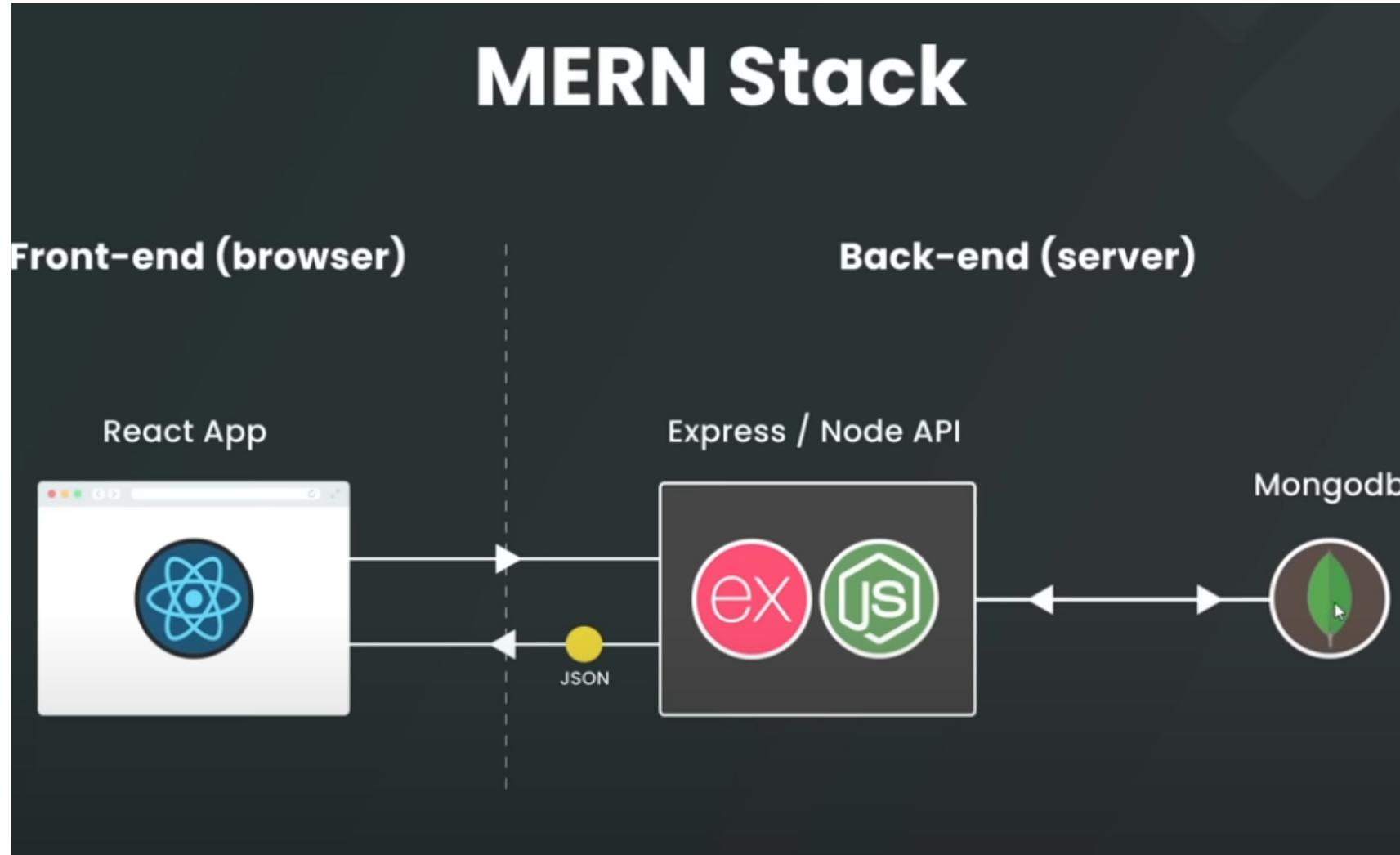


React



## Frontend: React

- provide client-side components for users to interact with the system



## Middleware: Express

- handle HTTP request and responses
- connection to MongoDB

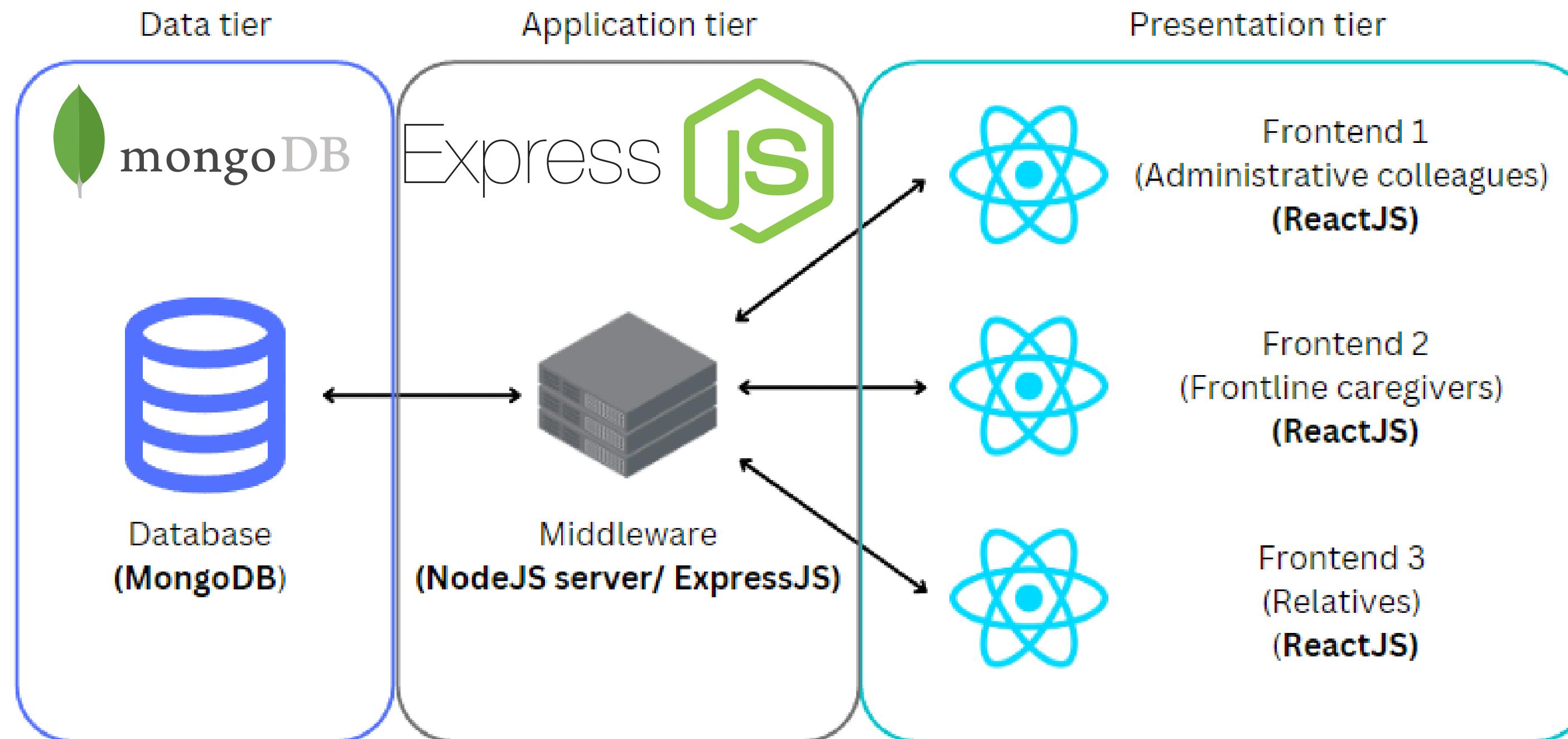
## Middleware: NodeJS

- runtime environment
- Node Package Manager (NPM) contains different packages for development

## Backend: MongoDB

- NoSQL database
- JSON format data flow

# Overall system structure

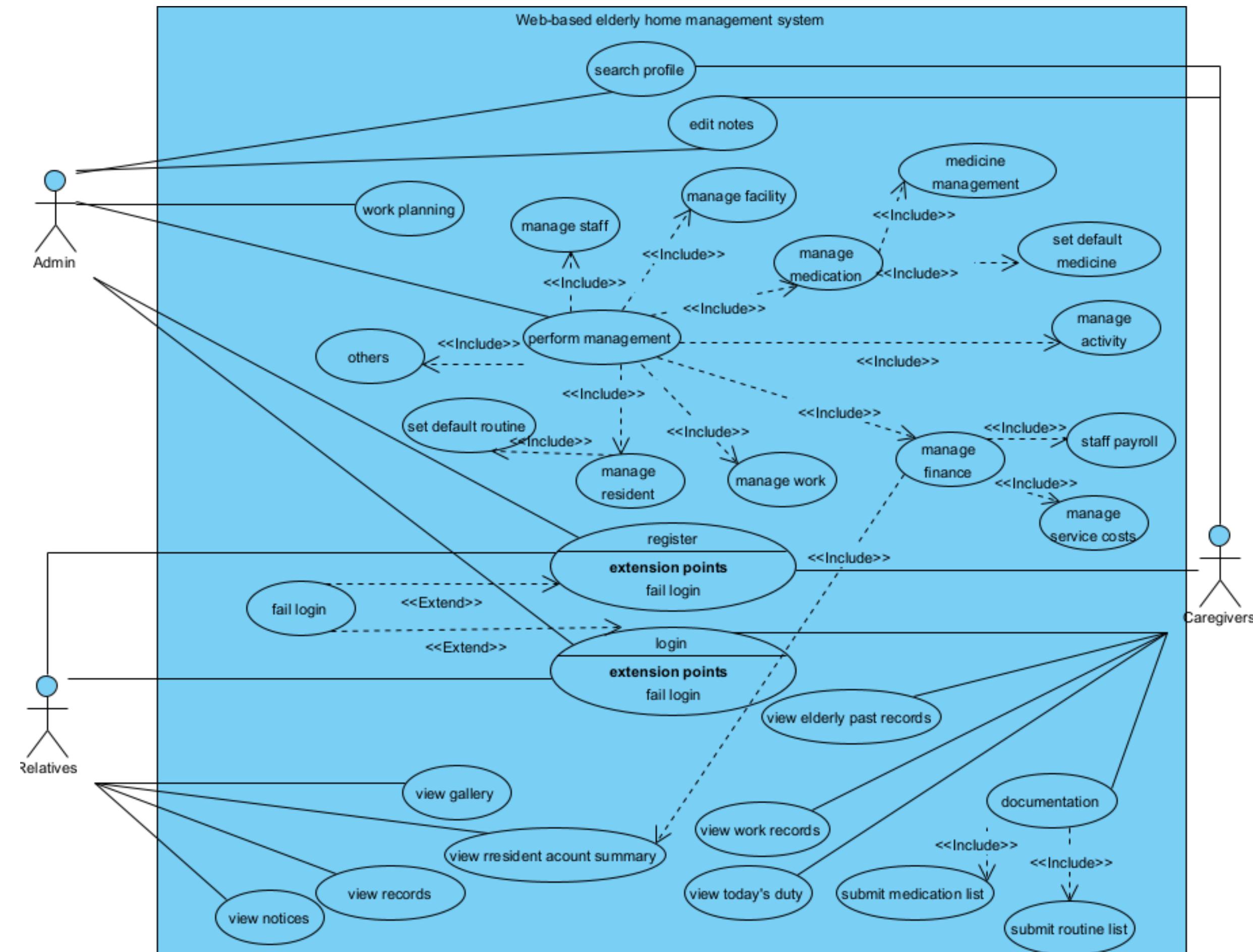


03

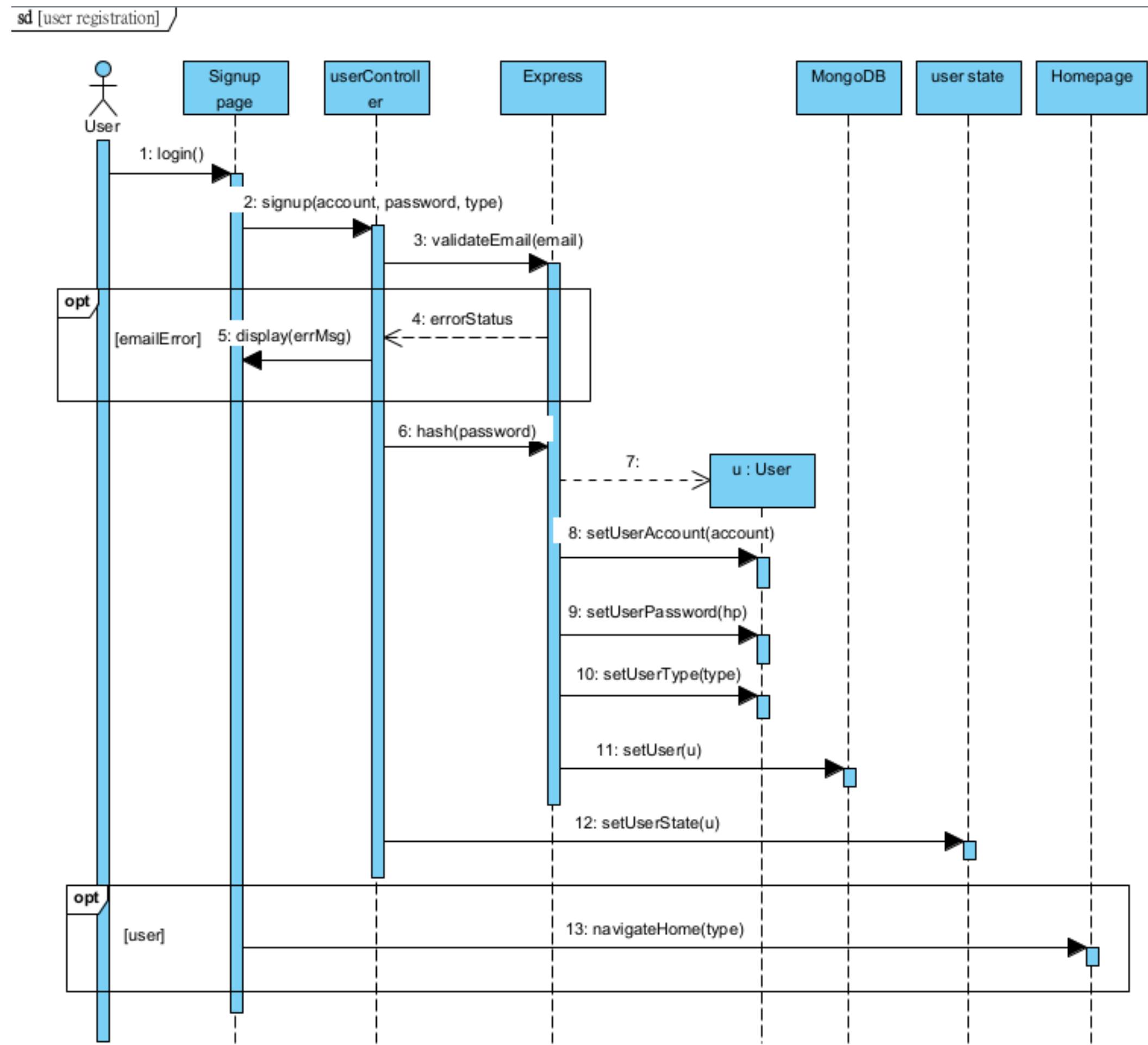
# UML diagrams



# Use Case Diagram

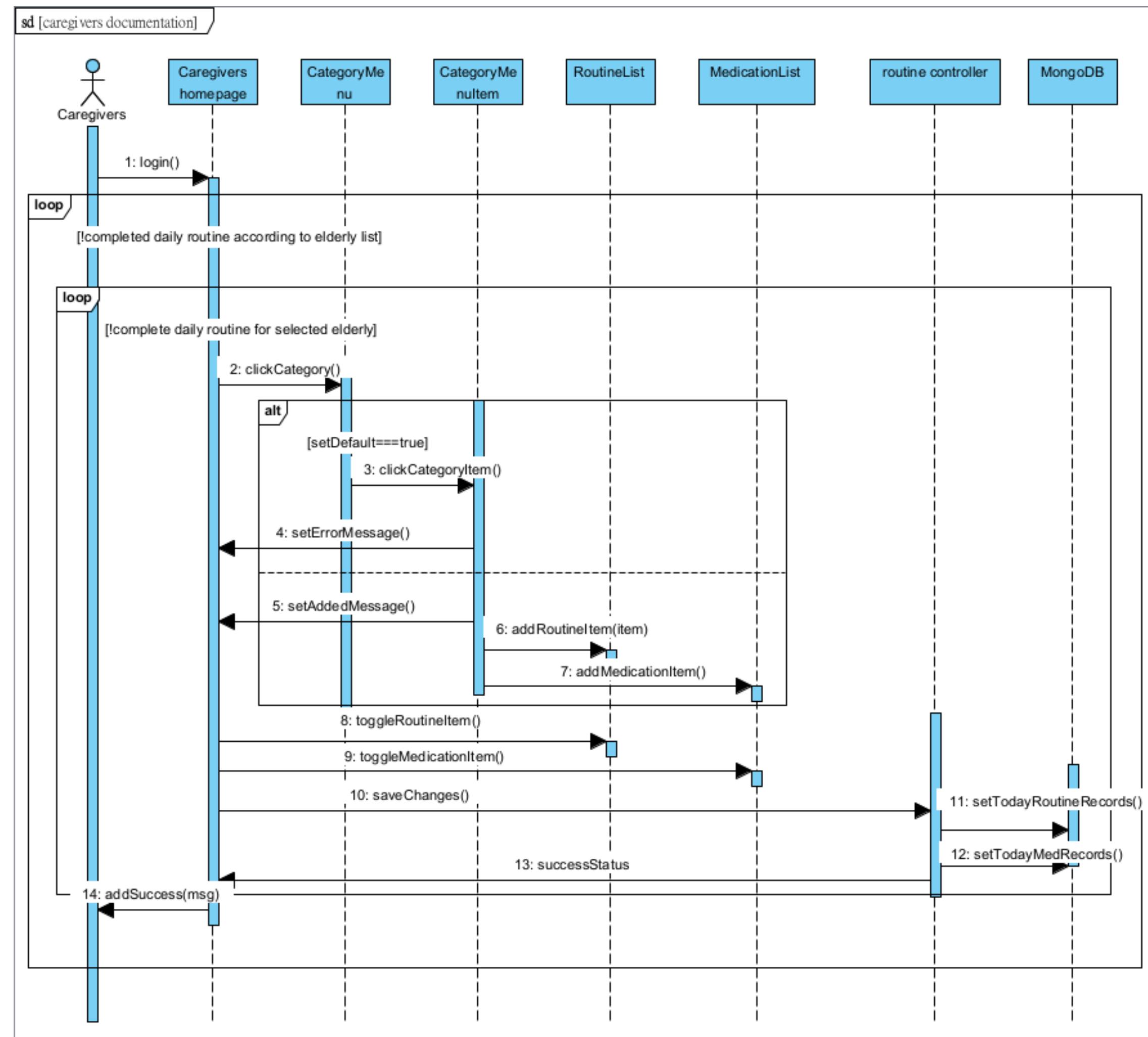


# Sequence Diagram (Registration: all users)



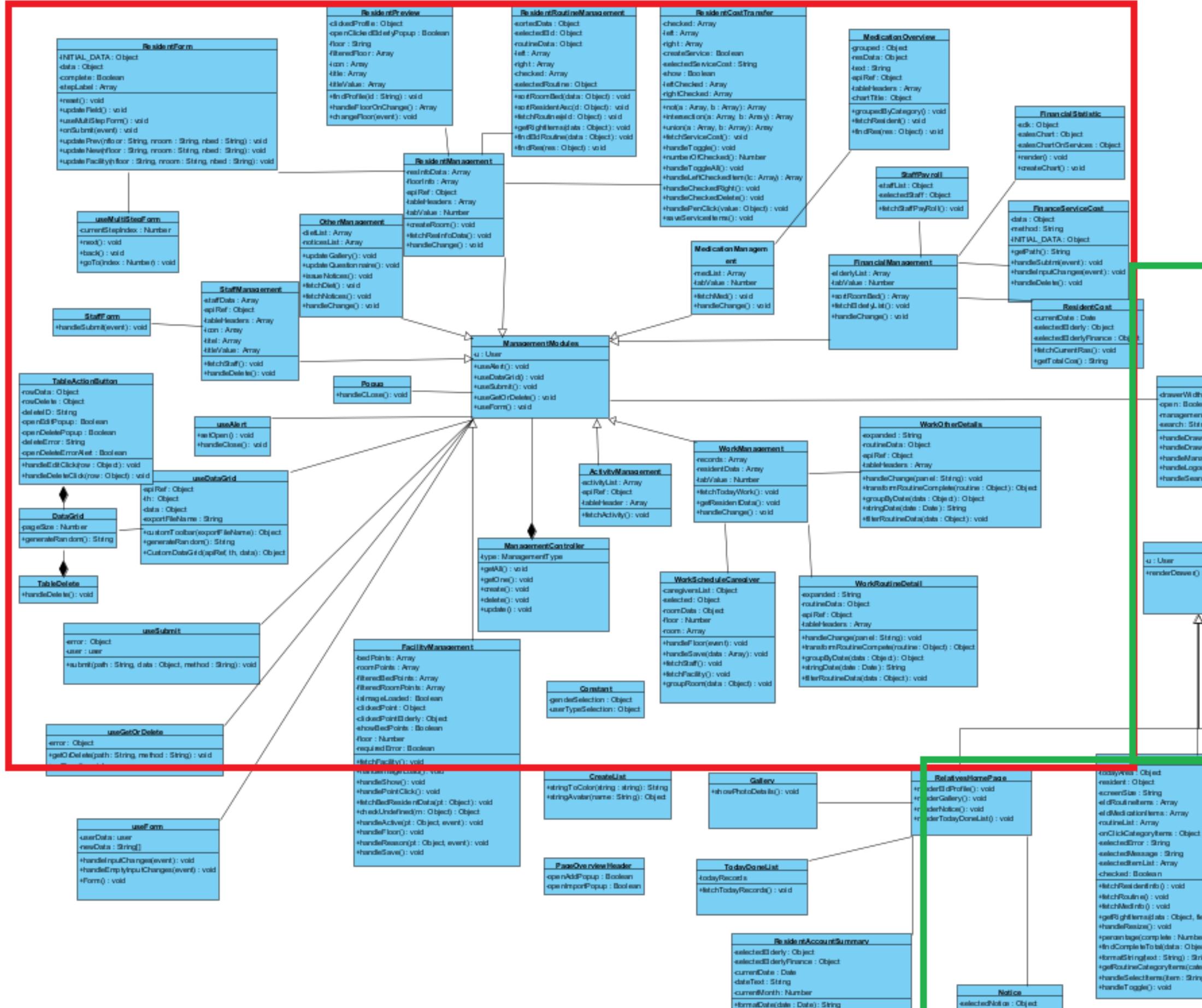
# Sequence Diagram

## (Documentation: Caregivers)

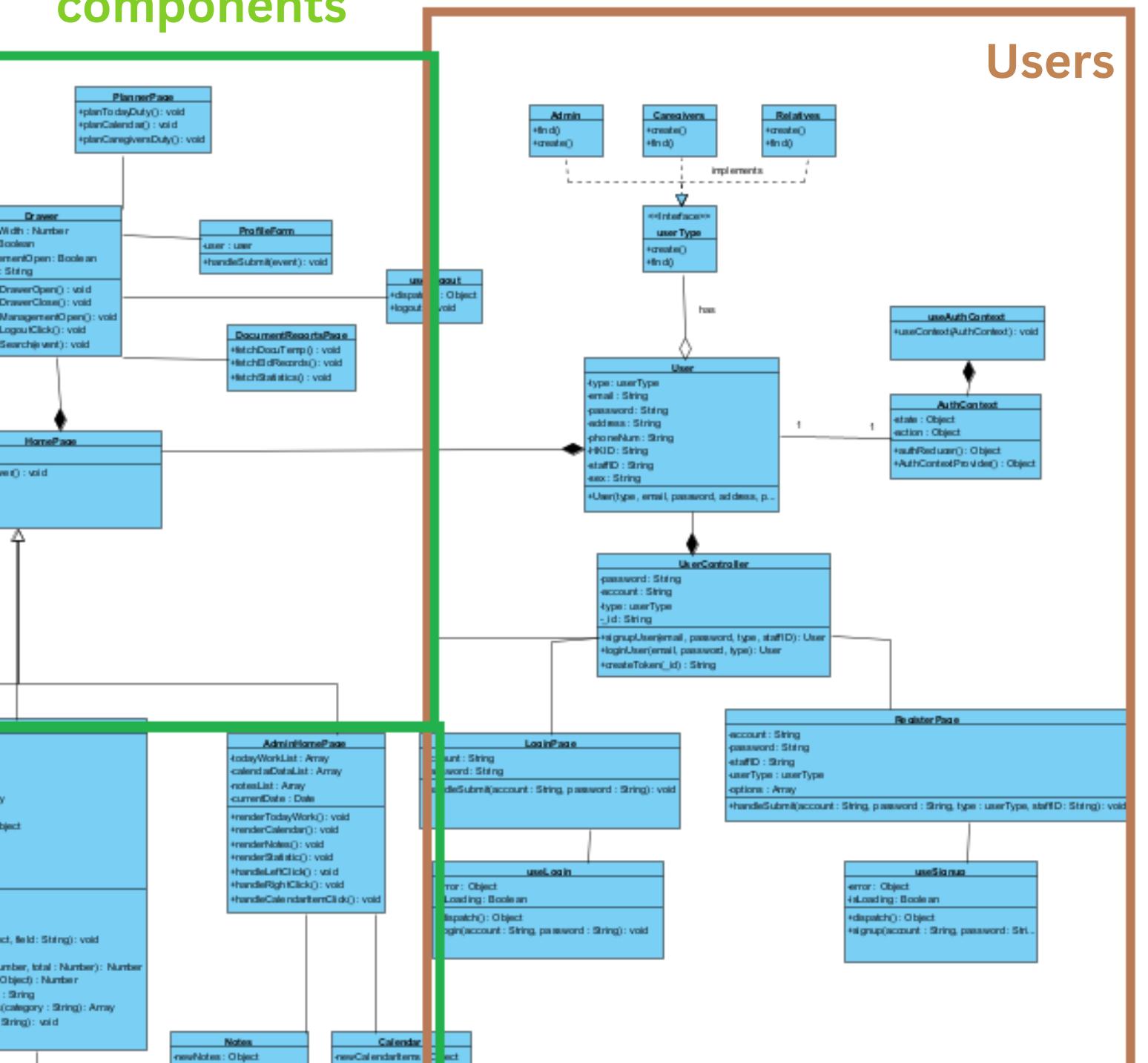


# Class Diagram

## Management modules

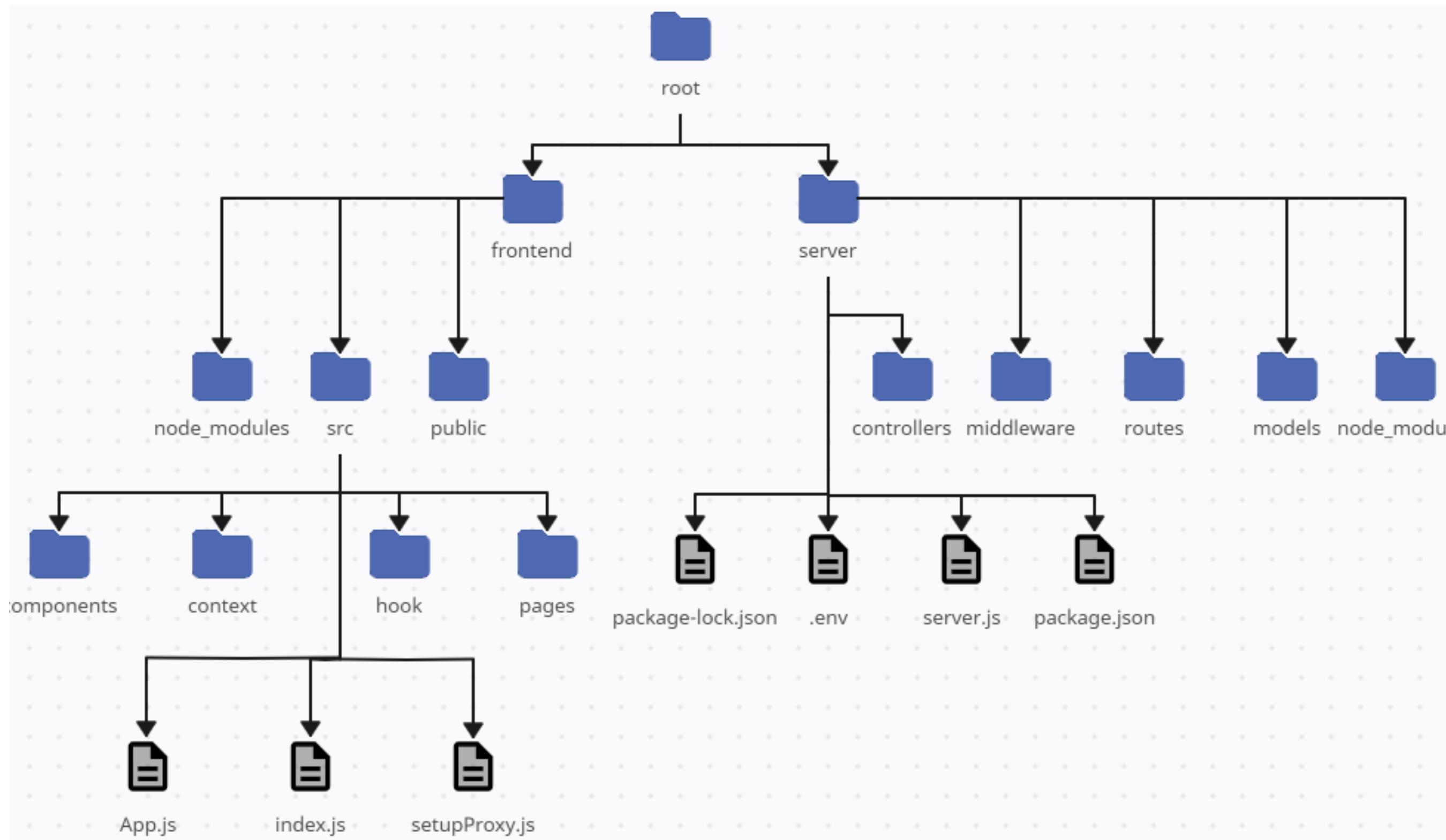


Homepage and common components



Users

# Directory and file structure





04

# Methodology & Implementation

1

# MongoDB

- Database Connection
- Database Trigger
- MongoDB charts

▼ test

activities

calendaritems

diets

facilities

galleries

medications

notes

notices

residentaccountsumm...

residentinfos

routines

servicescosts

todayworkrecords

users

○ PS D:\Matthew\University\FYP\system\backend> **npm run dev**

> backend@1.0.0 dev

> nodemon server.js

[nodemon] 2.0.20

[nodemon] to restart at any time, enter `rs`

[nodemon] watching path(s): .\*

[nodemon] watching extensions: js,mjs,json

[nodemon] starting `node server.js`

Connected to db and Listening on port 4000

/api/management/residents GET

/api/management/residents GET

/api/management/residents GET

/api/management/residents GET

/api/management/medication GET

/api/management/medication GET

/api/management/residents GET

/api/management/residents GET

□

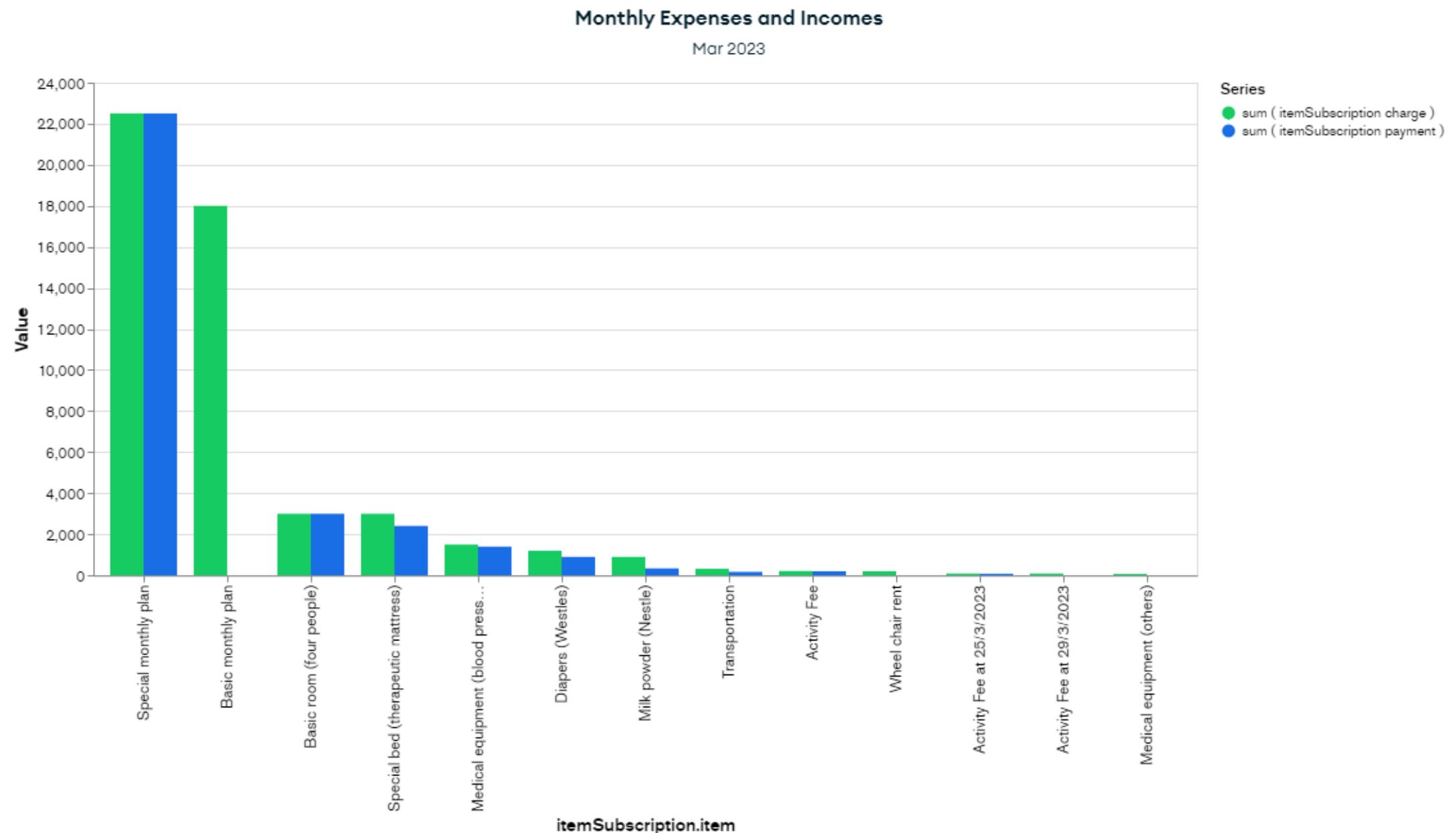
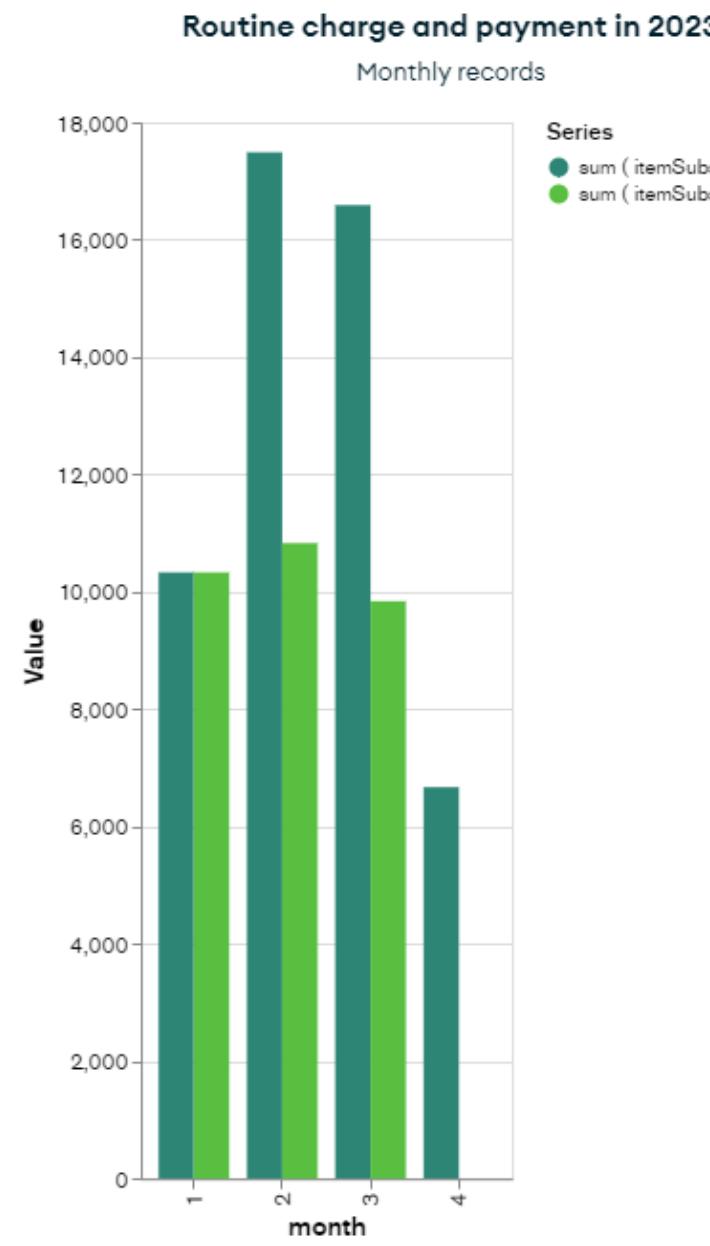
# Database Scheduled Triggers

## (Monthly-basis Resident Account Summary)

```
1  exports = function() {
2    const mongodb = context.services.get("FYP")
3    const ras = mongodb.db("test").collection("residentaccountsummaries")
4    const res = mongodb.db("test").collection("residentinfos")
5
6    const currentDate = new Date();
7    const prevMonth = currentDate.getMonth()
8    const currentMonth = currentDate.getMonth() + 1;
9    const currentYear = currentDate.getFullYear();
10   const lastMonthSecond = new Date(currentYear, currentMonth, 0, 23, 59, 59, 999)
11
12
13
14   function previousRas(id, month, callback) {
15     ras.findOne({$and: [{residentID: id}, {'month': month}]}), function(err, result) {
16       if (err) {
17         return console.error(err);
18       } else {
19         console.log(id, result)
20         return result ? result.itemSubscription : []
21       }
22     }
23   }
24
25   return res.aggregate([
26     // all residents
27     {$match: {}}
28   ]).toArray()
29   .then(residents => {
30     residents.forEach(resident => {
31       const accountSummary = {
32         residentID: resident.residentID,
33         month: currentMonth,
34         payDate: "",
35         deadlinePayData: lastMonthSecond,
36         payerName: resident.relativesName,
37         payerContact: resident.relativesPhone,
38         payerRelation: "family",
39         payType: "",
40         payAmount: "",
41         payDescription: "",
42         // default: previous month item subscription
43         itemSubscription: previousRas(resident.residentID, prevMonth),
44         transDate: "",
45         paid: false,
46         received: false
47       }
48       ras.insertOne(accountSummary)
49
50     })
51   })
52   .catch(
53     (err=> console.error("Failed to generate report for resident: ${resident.residentID} ", err))
54   )
55
56};
```

# MongoDB charts

## (Financial overview categorized based on month and items)



2

# ExpressJS

- Router: path routing  
**(backend)**

```

101 //WorkManagement
102
103 //1. TodayWorkRecords
104 router.post("/work/todayworkrecords", createTodayWork)
105 router.get("/work/todayworkrecords", fetchTodayWork)
106 router.get("/work/allworkrecords", fetchAllTodayWork)
107
108 //2. Routine
109 router.post("/work/routine", createRoutine)
110 router.get("/work/routine", fetchRoutine)
111 router.patch("/work/routine/:id", updateRoutine)
112 router.patch("/work/routine/updateAdd/:id", updateRoutine_Add)
113 router.patch("/work/routine/updateDel/:id", updateRoutine_Del)
114
115 //MedicationManagement
116
117 //1. Medication records
118 router.post("/medication/todaymedicationrecords", createTodayMedication)
119 router.get("/medication/todaymedicationrecords", fetchTodayMedication)
120 router.get("/medication/allmedicationrecords", fetchAllMedicationRecords)
121
122 //2. Medicine management
123 router.post("/medication", createMed)
124 router.get("/medication", fetchMed)

```

3

# ReactJS

- library for frontend development
- React-router: frontend routing  
**(frontend)**

```

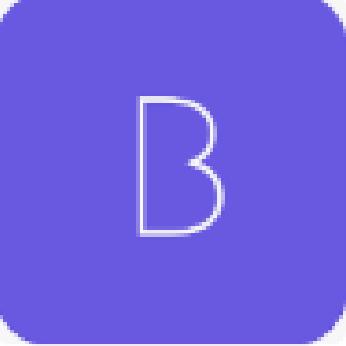
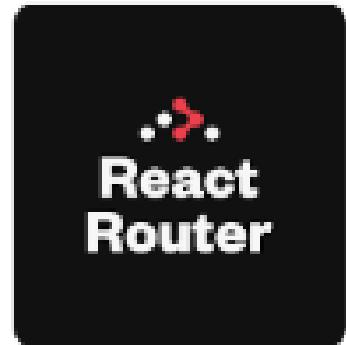
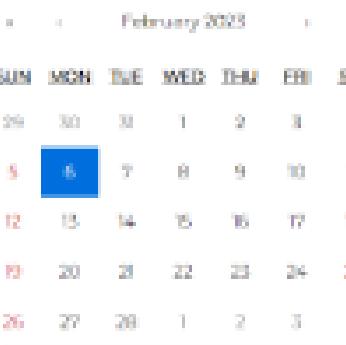
<BrowserRouter>
  <div className="pages">
    <Routes>
      <Route path="/home" element={user ? userHomePage(user.userType) : <Navigate to="/login"/>} />
      <Route path="/login" element={!user? <Login />: <Navigate to="/home"/>} />
      <Route path="/signup" element={!user ? <Signup/>: <Navigate to="/home"/>} />
      <Route path="/profile" element={user? <Drawer main={<Profile/>}>: <Navigate to="/login"/>} />
      <Route path="/management/staff" element={user? <Drawer main={<StaffManagement/>}>: <Navigate to="/login"/>} />
    </Routes>
  </div>
</BrowserRouter>

```

4

# Node JS

- Javascript runtime
- Node Package Manager

Development	Packages
Backend	  <b>mongoose</b>
Frontend	 

## 5

# Material UI

- library of components for React applications to boost development
- user-friendly and responsive interfaces

The screenshot shows a Material UI application for 'Kwun Tong On Tai Elderly Home'. The left sidebar has a yellow border and contains a navigation menu with items like Dashboard, Management modules, Staff Management, Residents Management, Work Management, Financial Management, Medication Management (which is selected and highlighted with a green border), Activity Management, Facility Management, Others, Planner, History and records, Documents and reports, and a red box around Notification and Settings.

The main content area has a blue header bar with the title 'Medication Management' and a search bar. Below the header is a sub-header with icons for medicine bottles and a search bar. A green box highlights the tabs: Medication Overview, Medical Appointment, Residents Medication Management, and Medical Records.

A large section displays a pie chart titled 'Total number of drugs' showing 421 total medicines, with 69.3% being Oral. To the right is a search bar and a message: 'No medicine found' followed by a note about OpenFDA and a link to add a new medicine.

At the bottom, a red box highlights a table titled 'Selected Medicine: Abacavir'. The table has columns for Manufacturer, Product usage, Substance, and Actions. It lists several entries, with one row for 'Abacavir sulfate' highlighted in purple.

Manufacturer	Product usage	Substance	Actions
Cipla usa inc.	Oral	Abacavir sulfate	<span style="color: purple;">Edit</span> <span style="color: purple;">Delete</span>
Mylan pharmaceuticals inc.	Oral	Abacavir sulfate	<span style="color: purple;">Edit</span> <span style="color: purple;">Delete</span>
U.S. pharmaceuticals inc.	Oral	Abacavir sulfate	<span style="color: purple;">Edit</span> <span style="color: purple;">Delete</span>
B. braun	Intravenous	Acetaminophen	<span style="color: purple;">Edit</span> <span style="color: purple;">Delete</span>

Different pre-defined components to boost development

**MUI** Search elderly profile...

**Today's Working Area**  
Shift: A  
Room: 101,105,106

**Routine Progress**  
0%  
Completed: 0  
Not complete: 7

Name: Chan, Tsz (Resident ID: 12)  
Bed: 101-A  
Special Routine Items: 4  
Default Medication Items: 1

**Completed Routine Items** 0    **Completed shift medication** X

**View routine records for elderly Chan, Tsz**  
**View medication records for elderly Chan, Tsz**

**Default**

**Routine List**

Default Routine Items

- ★ Changing incontinence products
- ★ Feeding
- ★ Blood glucose measurement
- ★ Blood pressure measurement

Additional Routine Items  
SAVE CHANGES

**Medication List**

**FEEDING** **BLOOD PRESSURE MEASUREMENT** **BLOOD GLUCOSE MEASUREMENT**  
**WEIGHT** **OXYGEN SATURATION** **URINE OUTPUT**  
**BODY TEMPERATURE** **RESPIRATORY RATE** **HEART RATE**  
**WOUND CARE** **HEIGHT** **URINARY CATHETER CARE**  
**HANDLE PRESSURE SORE** **MEDICAL CARE** **MENTAL HEALTH CARE**  
**REHABILITATION THERAPY** **SKIN CARE** **ORAL CARE**

**Today's Working Area**  
Shift: A  
Room: 101,105,106

**Routine Progress**  
0%  
Completed: 0  
Not complete: 7

**Duty Today**

CT Resident ID: 12 Name: Chan, Tsz Bed: 101-A	Resident ID: 89 Name: Chan, Tsz Ching Bed: 101-B
TT Resident ID: 50 Name: Tai, Tai Bed: 101-C	TM Resident ID: 13 Name: Tai Man, Chan Tia Bed: 105-A
CG Resident ID: 28 Name: Chan, Gong Gong Bed: 105-B	CA Resident ID: 15 Name: c, a Bed: 105-C
GN Resident ID: 18 Name: Good, Night Bed: 105-D	

Name: Chan, Tsz (Resident ID: 12)  
Bed: 101-A  
Special Routine Items: 4

Responsive design for caregivers  
(achieved by Grid component in Material UI)

6

## Fullcalendar

- virtual calendar/ list to display and view important events
- save single and recurring events

### Work schedule

< > today

April 9 – 15, 2023

List Day List Week List Month

Sunday	April 9, 2023
8:00am	● Matthew, Cas - 101, 102
10:00pm	● Chan, Grid now - 201, 202, 203
Monday	April 10, 2023
8:00am	● Chan, Grid now - 101, 103, 105
8:00am	● Chan, Lam tsz - 101, 105, 106
2:00pm	● ASD, Casd - 101, 102, 103, 104, 105

Display **working area and shift records** in a list  
(Work schedule: Work management)

## Activity Overview

[Add an activity](#)[today](#) [<>](#)

April 2023

[month](#) [week](#) [day](#)

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	26	27	28	29	30	31
	2	3	4	5	6	7
	9	10	11	12	13	14
3p Visit from Cityu - Volunteer	● 8:30a Walking Training - Rehab ● 12p Book sharing (English) - Education ● 3p Small group gathering - Others	● 8:30a Walking Training - Rehab ● 12p Book sharing (English) - Education ● 3p Small group gathering - Others	● 2a Knitting - Handicrafts	● 2:30p Board game (Rummikub) ● 3:30p Board game (Mahjong) - I ● 4:20p Exercise classes - Rehab	● 1p Lunch sharing - Others ● 3p Visit from Cityu - Volunteer +2 more	● 2:20p Book sharing - Education ● 4a Recurring - Recreational
	16	17	18	19	20	21
● 12p Staff Development Day - Others	● 8:30a Walking Training - Rehab ● 2a Knitting - Handicrafts	● 8:30a Walking Training - Rehab ● 2a Knitting - Handicrafts	● 2:30p Board game (Rummikub) ● 3:30p Board game (Mahjong) - I ● 4:20p Exercise classes - Rehab		● 12p Visiting from City University ● 4a Recurring - Recreational	
23	24	25	26	27	28	29
	8a staff meeting - Others ● 8:30a Walking Training - Rehab ● 2a Knitting - Handicrafts ● 9a Talks (Death) - Educational		● 2:30p Board game (Rummikub) +2 more			

Display **activities schedule** in a virtual calendar  
(Activity management)

05

# Algorithm Design

```
state={  
  products: storeProducts  
}  
  
render() {  
  return (  
    <React.Fragment>  
      <div className="py-5">  
        <div className="container">  
          <Title name="our" title="Our Products" />  
          <div className="row">  
            <ProductConsumer>  
              {(value) => {  
                console.log(value);  
              }}  
            </ProductConsumer>  
          </div>  
        </div>  
      </div>  
    </React.Fragment>  
  );  
}
```

1

# Shorten searching elderly information

- Resident Overview
- Searching options
- Datagrid

The screenshot displays a resident management application interface. At the top, there is a search bar labeled "Search elderly..." and a dropdown menu for "Floor" set to "All Floor". Below this, a grid of elderly profiles is shown, each with a small illustration of a resident and their details:

Bed	Name	Bed	Name	Bed	Name	Bed	Name	Bed	Name	Bed	Name
A	Chan, Tsz Ching	B	Chan, Tsz	C	Tai, Tai	A	ASD,asd	B	C, AD	D	C, A
A	Number, 9	B	Chan, Tai	C	Ta, Frst	B	Chi Chi, Lam	C	Chan, Asd	D	Chan, BOBO

Below the grid, a modal window titled "Elderly profile" is open, showing three tabs: "Elderly Information", "Relatives Information", and "Room Information". The "Elderly Information" tab contains the following fields:

Field	Value
Resident ID	89
Last Name	Chan
First Name	Tsz Ching
Gender	M
HKID	Y9999115
Active	Yes
Age	80
Height (cm)	150
Weight (kg)	20
Address	Address

## Residents Overview

[Import](#)
[Add new residents](#)
[Resident Overview](#)
[Resident Details](#)
[Resident Routine Management](#)
[Routine Records](#)

SEARCH BY														COLUMNS			EXPORT										
Columns	Operator	Value	Age				Height (cm)			Weight (kg)		Address		Relatives Name		Relatives Phone		Relatives HKID		Relatives Address		Relatives Email		Floor		Zone	Room
X Resident ID	contains	▼ Filter value																									
8	Chan	Asd	X	M	Y8402684	70	150	50	HelloHongKong	Ctn	*	*	*	*	*	*	*	*	*	*	*	1	A	104			
9	Number	9	✓	M	Y8056165	70	150	50	HelloHongKong	on nine nine	99999999	Y8949999	ahodishaoidhaiohda Long, Hong Kong, jeqiowejqoijeqoiwje q	GG@yahoo.com.hk	1	A	103										
10	OHAOISH D	AOIHDOIA D	X	M	y8408	70	150	50	HelloHongKong	Chan Tai Man, Tai Chi YE	98489481/84894848	Y6806106	ADDRESS	EMAIL	2	A	201										
11	ASD	asd	✓	N/A	y40464	70	150	50	HelloHongKong	/	/	/	/	/	/	/	/	/	/	1	A	102					
12	Chan	Tsz	✓	N/A	Y808616	70	150	50	HelloHongKong	Chan, Tao Tao	95951616	/	/	/	/	/	/	/	/	1	A	101					
13	Tai Man	Chan Tia	✓	F	Y8016123	70	150	50	HelloHongKong	CCCCHAHAAA LOREM	123910389	G8186186	Kam Tai hOuse, On Tai ESTATE, Kwun Tong, Kowloon	email	1	A	105										
14	c	A	✓	F	g9999888	70	150	50	HelloHongKong	/	/	/	/	/	/	/	/	/	1	B	114						
15	c	a	✓	M	y46846846	70	150	50	HelloHongKong	Tai Man Chan	9998888	Y80864608	address	email	1	A	105										
16	LN	FN	✓	F	G9999888	80	150	60	ADDRESS	ChanTaiMan	984894	6065156	ASDASDAS	email	3	A	301										
17	Lam	Tai Man	✓	M	Y8065156	90	150	50	Address	Relatives Name is so long	19238190	Y604655	Address	Email@gmail.com	3	B	313										
18	Good	Night	✓	N/A	Y11111	80	90	90	Address	Chan Tai Man	9595959595	Y805151	Address	Email	1	A	105										

DataGrid for storing all elderly information with **filtering**

2

## Shorten documentation process

- Today's List and Today's working area
- Default routine and medication items

### Today's Working Area

*Shift: A*

*Room: 101,105,106*

### Duty Today



Resident ID: 12  
Name: Chan, Tsz  
Bed: 101-A



Resident ID: 89  
Name: Chan, Tsz Ching  
Bed: 101-B



Resident ID: 50  
Name: Tai, Tai  
Bed: 101-C



Resident ID: 13  
Name: Tai Man, Chan Tia  
Bed: 105-A



Resident ID: 28  
Name: Chan, Gong Gong  
Bed: 105-B



Resident ID: 15  
Name: c, a  
Bed: 105-C



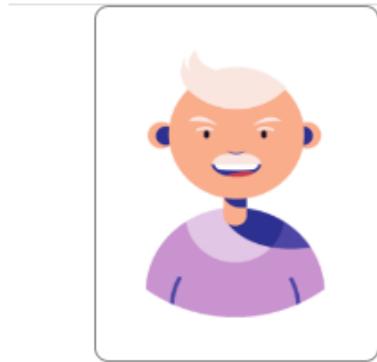
Resident ID: 18  
Name: Good, Night  
Bed: 105-D

Provide lists of elderly for health care

# Admin

## Residents Overview

Resident Overview Resident Details Resident Routine Management Routine Records



101-A | N/A | 70 | Chan, Tsz

101-B | M | 80 | Chan, Tsz Ching

101-C | M | 80 | Tai, Tai

102-A | N/A | 70 | ASD, asd

102-B | F | 70 | C, AD

102-D | F | 50 | C, A

103-A | M | 70 | Number, 9

103-B | F | 70 | Chan, Tai Man

Name: Chan, Tsz (Resident ID: 12)

Room: 101, A

Relative Name: Chan, Tao Tao Relative Contact: 95951616

Today's Routine Items  
Default Routine Items

Total Routines  
0/15 selected

- Changing incontinence products
- Toileting
- Feeding
- Blood pressure measurement
- Blood glucose measurement
- Oxygen saturation
- Respiratory rate

Default Routines to Chan, Tsz  
0/4 selected

- Changing incontinence products
- Feeding
- Blood glucose measurement
- Blood pressure measurement

Finish Today Routines



Default Routine Items

4

### Routine Information

Routine Name Default routine to elderly

Changing incontinence

Default To  
12, 11, 6

Routine Category

cleaning (elderly)

Resident ID: 12  
Name: Chan, Tsz  
Bed: 101-A

Fixed Time

No

Resident ID: 11  
Name: ASD, asd  
Bed: 102-A

Resident ID: 6  
Name: C, AD  
Bed: 102-B

# Caregivers

## Default

### Routine List

#### Default Routine Items

★ Changing incontinence products

★ Feeding

★ Blood glucose measurement

★ Blood pressure measurement

#### Additional Routine Items

**SAVE CHANGES**

Set elderly **default routine items** (e.g. special needed)

3

# Optimization: JSON web tokens

- standard, industry-wised security technique to protect react routes by authorization

```
const jwt = require("jsonwebtoken")
const User = require('../models/userModel')

const requireAuth = async(req, res, next) => {

    // verify authentication
    const { authorization } = req.headers

    if(!authorization){
        return res.status(401).json({error: 'Authorization token required'})
    }

    const token = authorization.split(' ')[1]
    //console.log(token)
    try {
        // when login, it will generate the token -> then find the user with token
        const {_id} = jwt.verify(token, process.env.SECRET)
        // attach user object in requests
        req.user = await User.findOne({ _id }).select('_id')
        // allow move on to next function
        next()
    } catch (error) {
        console.log(error)
        res.status(401).json({error: 'Request is not authorized'})
    }
};

module.exports = requireAuth
```

```
1
2   "error": "Authorization token required"
3 }
```

*No request bearer tokens*

The screenshot shows a Postman request for a login endpoint. The request body is a JSON object with 'email' and 'password' fields. The response status is 200 OK, and the response body is a JSON object containing an 'email' field and a 'token' field. A tooltip for the 'token' field indicates it will be automatically generated when the request is sent.

```
1
2   "email": "helloworld@gmail.com",
3   "password": "Hgps3b32"
4 }
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```
1
2   "email": "helloworld@gmail.com",
3   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2M2UwODNlNjdjNmRlM..."
```

Type Bearer T... Token eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2M2UwODNlNjdjNmRlM...

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#).

Status: 200 OK Time: 10 ms Size: 2.11 KB Save Response

```
1
2   "error": "Request is not authorized"
3 }
```

*Token is not in correct format*

The screenshot shows a Postman request for a login endpoint. The request body is a JSON object with 'email' and 'password' fields. The response status is 200 OK, and the response body is a JSON object containing user profile information like '\_id', 'email', 'password', 'address', etc.

```
1
2   "HKID": "",
3   "staffID": "",
4   "__v": 0
5
6
69   "HKID": "",
70   "staffID": "",
71   "__v": 0
72
73   {
74     "_id": "63dfda83b6cc789d8cbf40f1",
75     "email": "testwithmoreparameter2@gmail.com",
76     "password": "82b$10$P6xxmc/QelNLfoOB8WPN.eHb3T4SdZajfAScVrRmTsX/nAk78oeAe",
77     "address": "",
78     "phoneNum": "",
79     "sex": "",
80     "HKID": "",
81     "staffID": "",
82     "__v": 0
83 }
```

***Create tokens after login, successful in fetching data***



06

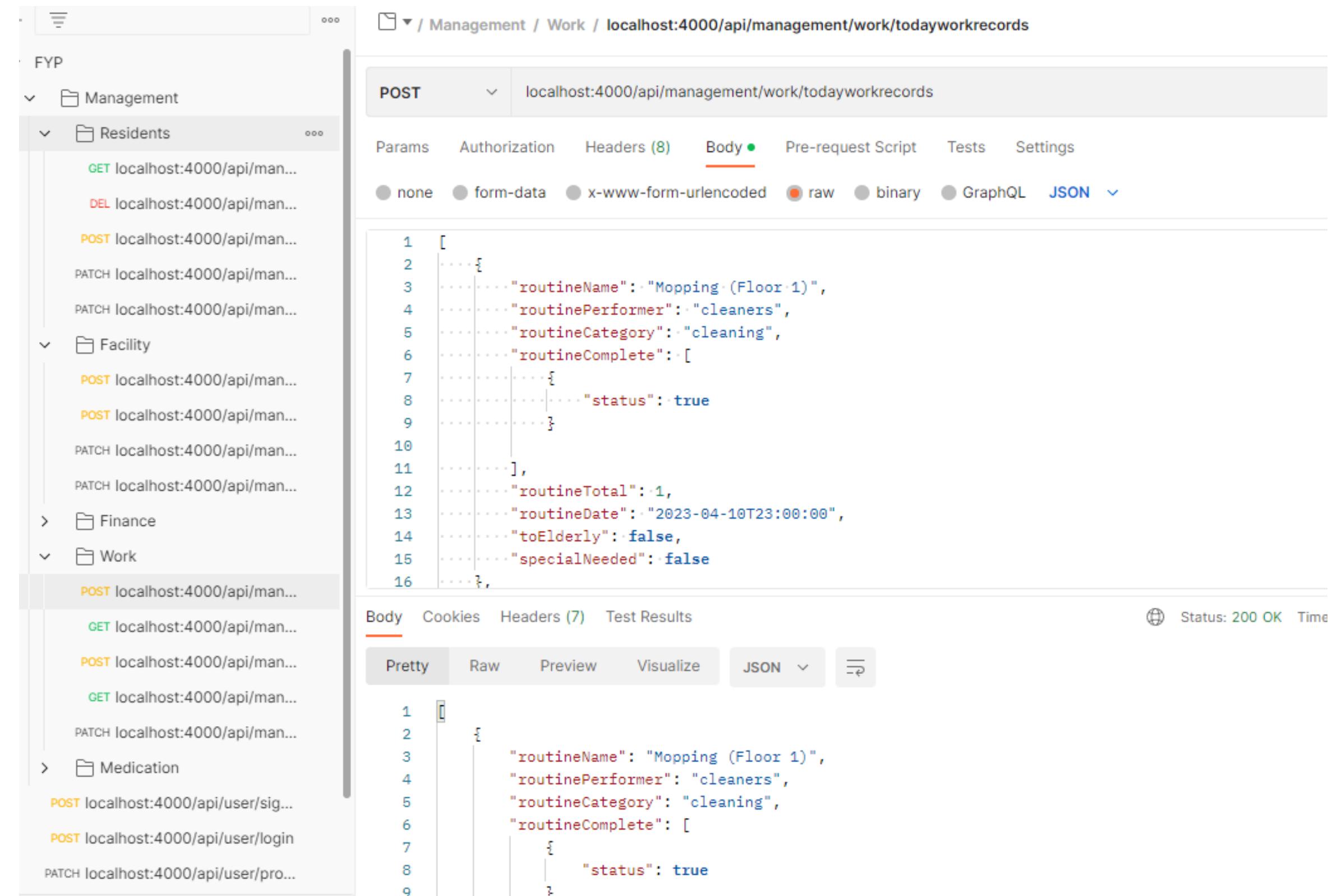
# Testing & Evaluation

1

# Route testing

- **backend** testing (HTTP request)
- Testing environment:  
**Postman**

100%



The screenshot shows the Postman application interface. On the left, there is a sidebar with a tree view of API endpoints under the 'FYP' project. The 'Work' section is expanded, showing various methods like POST, GET, PATCH, and DELETE for different URLs. On the right, a specific POST request is selected for the URL `/Management/Work/localhost:4000/api/management/work/todayworkrecords`. The 'Body' tab is active, showing a JSON payload:

```
[{"routineName": "Mopping (Floor 1)", "routinePerformer": "cleaners", "routineCategory": "cleaning", "routineComplete": [{"status": true}], "routineTotal": 1, "routineDate": "2023-04-10T23:00:00", "toElderly": false, "specialNeeded": false}]
```

Below the body, the response status is shown as `Status: 200 OK`. The response body is also displayed in JSON format:

```
{"routineName": "Mopping (Floor 1)", "routinePerformer": "cleaners", "routineCategory": "cleaning", "routineComplete": [{"status": true}]} 
```

2

# Logic testing

- **function** testing  
(data calculation, string matching, array matching)
- Testing environment:  
**JEST**



src/tests/ras_test.test.js					
File	% Stmt	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	97.77	92.85	100	97.61	
ras_test.js	100	100	100	100	
work_schedulecaregivers_test.js	96.42	90	100	96.15	6

```

PASS  src/tests/ras_test.test.js
-----|-----|-----|-----|-----|-----|
File    | % Stmt | % Branch | % Funcs | % Lines | Uncovered Line #s
-----|-----|-----|-----|-----|-----|
All files | 97.77 | 92.85 | 100 | 97.61 | 
          | 100 | 100 | 100 | 100 | 
          | 96.42 | 90 | 100 | 96.15 | 6
-----|-----|-----|-----|-----|-----|
Test Suites: 1 failed, 1 passed, 2 total
Tests:       3 failed, 11 passed, 14 total
Snapshots:   0 total
Time:        0.835 s, estimated 1 s
Ran all test suites.

```

```

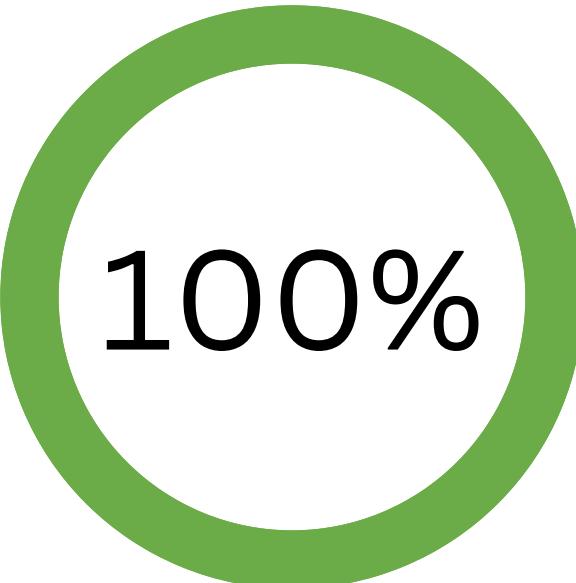
JS work_schedulecaregivers_test.test.js U ✘
frontend > src > tests > JS work_schedulecaregivers_test.test.js > ⚙️ test("gather events correctly - 1: no, 2: 2, 3: 1")
1  const schedulecg = require("./work_schedulecaregivers_test");
2
3  test('getColor: "", () => {
4    | expect(schedulecg.getColor("T08:00:00")).toMatch("green");
5  );
6  test("getColor: green", () => {
7    | expect(schedulecg.getColor("T14:00:00")).toMatch("");
8  );
9  test("getColor: brown", () => {
10   | expect(schedulecg.getColor("T15:00:00")).toMatch("brown");
11 );
12
13 test("getDate: only date", () => {
14   | expect(schedulecg.getDate("2023-04-05T18:00:00")).toMatch("2023-04-05");
15 );
16 test("getDate: datetime is wrong", () => {
17   | expect(schedulecg.getDate("2023-04-05T18:00:00")).not.toMatch(
18     | "2023-04-05T18:00:00"
19   );
20 );
21
22 test("gather events correctly - 1: no, 2: 2, 3: 1", () => {
23   | const data = [
24   | ];
25   | const result = [
26   | ];
27   | expect(schedulecg.gatherEvents(data)).toEqual(result)
28 );
29
30 });
31
32 });
33
34 });
35
36 });
37
38 });
39
40 });
41
42 });
43
44 });
45
46 });
47
48 });
49
50 });
51
52 });
53
54 });
55
56 });
57
58 });
59
60 });
61
62 });
63
64 });
65
66 });
67
68 });
69
70 });
71
72 });
73
74 });
75
76 });
77
78 });
79
80 });
81
82 });
83
84 });
85
86 });
87
88 });
89
90 });
91
92 });
93
94 });
95
96 });
97
98 });
99
100 });
101 });
102 });
103 });
104 });
105 });
106 });
107 });
108 });
109 });
110 });
111 });
112 });
113 });
114 });
115 });
116 });
117 });
118 });
119 });
120 });
121 });
122 });
123 });
124 });
125 });
126 });
127 });

```

3

## UI testing

- **interfaces** testing
- Testing method:  
**Test Plan**



100%

Test ID	Test Name	Status	Test Date
1	User login with incorrect data	✓	2/2/2023
2	User signup	✓	2/2/2023
3	Add new staff	✓	2/2/2023
4	Add new residents	✓	14/2/2023
5	Residents changes in resident overview	✓	14/2/2023
6	Set default routine items to elderly	✓	14/2/2023
7	Update services cost	✓	14/2/2023
8	Change 2/F Activity Room 1 to inactive	✓	3/3/2023
9	Schedule caregivers	✓	12/3/2023

# Caregivers

Test ID	Test Name	Status	Test Date
1	Correct today working area and shift		12/3/2023
2	Correct elderly list		12/3/2023
3	View resident routine record		12/3/2023
4	View resident medication record		12/3/2023
5	Save today's resident routine		15/3/2023
6	Save shift medication		15/3/2023
7	Category Selection		12/3/2023
8	Item spreading		15/3/2023

# Relatives

Test ID	Test Name	Status	Test Date
1	View account summary		14/3/2023
2	View resident routine record		14/3/2023
3	View resident medication record		14/3/2023
4	Click on and view gallery		14/3/2023
5	Click on and view notices		14/3/2023

07

# Conclusion



# In this project...



**TRANSFORM THE PAPER-BASED MANAGEMENT IN ELDERLY HOMES**



**REDUCE POTENTIAL DOCUMENTATION ERRORS**



**OPTIMIZE WORKFLOW EFFICIENCY AND CARING SERVICES PERFORMANCE**

1

## Goal #1

# Transform the paper-based management in elderly homes

### WHERE WE'RE AT

- Limited management systems specific to elderly health sector in Hong Kong
- Most of the documentation are still manually written records

### ACTION ITEMS

- ✓ Build a **computerized** web-based elderly home management system
- ✓ Electronic health records (**EHR**)
- ✓ **Database** for data storage and retrieval to cope with large demand in the future

2

## Goal #2

# Reduce potential documentation errors

### ACTION ITEMS

 Drop-down **selections/predefined terminologies** are supported

 **History logs** (Medication records, routine records, Database log) for error tracing

3

## Goal #3

# Optimize workflow efficiency and caring services performance

### ACTION ITEMS

 **Visual elements** are supported to enhance efficiency

 **Default** management system design

 Virtual **list/calendar/tables** for better visualization of data

 **Management modules** for systematic management

# Future improvements



- *Expand to different system structures  
(mobile application: iOS, Android...  
cloud services)*
- 

- *Improvement in achieving responsive web design*
- 

- *Improvement in more in-depth testing  
(JEST testing coverage & develop more test cases)*

Thank  
you