MORE SYSTEM ANALYSIS
OF AGAPE'S SYSTEM

Lovely Fernandez | ZEDEV_
3RD YEAR COMPUTER SCIENCE INTERNATIONAL | TU858-3

Table of Contents

More System Analysis

This report consists of 4 use cases which includes a narrative describing the given scenario analysis from Agape's system, as well as, 2 to 3 exception/alternative flows of each use case.

This report is an extension to Agape.
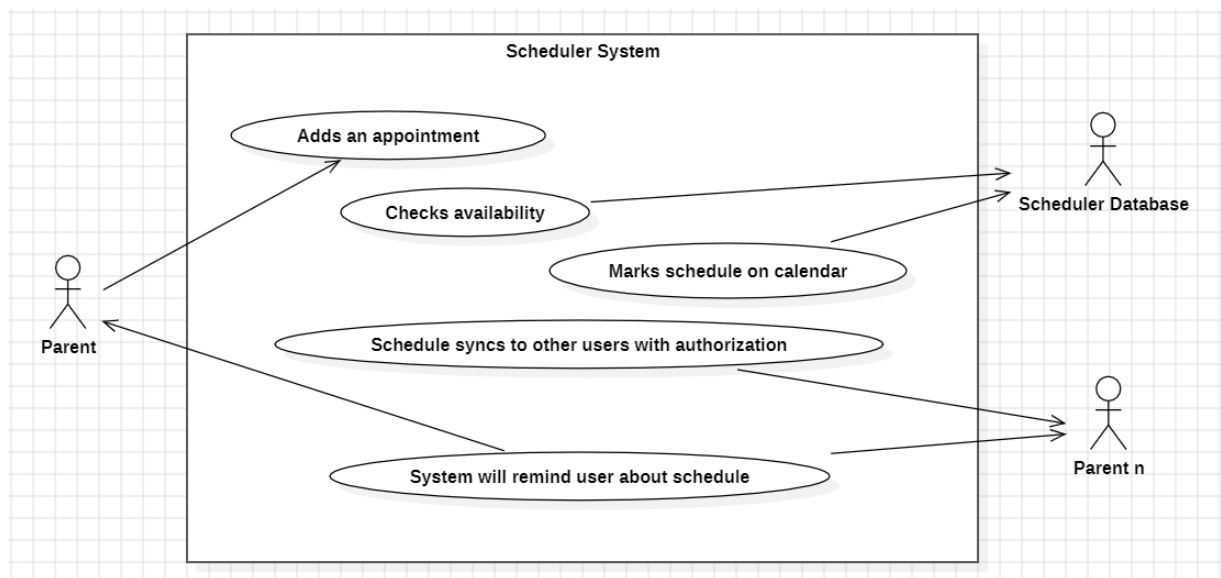
Use Case 2: Scheduler System

Figure 1: Use Case Diagram of Scheduler System (StarUML)

USE CASE	2	Using Scheduler System
Description of Goal in Context	User adds an appointment on the app	
Preconditions	To access scheduler system, a profile for the child must be added.	
Post Conditions, Success End Condition	After using Scheduler System, user has succesfully added an appointment on the calendar. Users connected to the child's profile will be updated and will have access to this appointment.	
Post Conditions, Failed End Condition	Appointment will not be saved	
Actors	Parent	

Trigger	User clicks adds a new appointment in scheduler section.
DESCRIPTION	The user creates an appointment which includes time, place and purpose of the appointment. This will be saved in the system and synced into relevant users.

Main Flow (MF)		
Step	Action	Alternate
2.1	User clicks (+) to add an appointment	
2.2	User inputs appointment details; title, description, time and date.	
2.3	System will check availability of the given timeslot and date.	EF.2.3
2.4	User saves the appointment.	
2.5	System updates its calendar.	
2.6	System requests user to set the reminder about the appointment.	
2.7	User will enter the time, date and duration.	
2.8	System will notify all relevant users of the new appointment and sync to their account calendar including the user's name of who created the appointment.	
2.9	On the day of the reminder, system will notify all users about the appointment.	
2.10	On the day of the appointment, system will notify all users about the appointment in the morning. (6AM)	
2.11	End use case.	
n...	<...>	

EXCEPTIONS or ERROR Flow Description		
	EF.2.3	
	Error flow at MF.2.3 : Timeslot and date have been booked.	
Step	Branching Action	Alternate
2.3.1	System notifies user about existing appointment on the given date and time.	
2.3.2	User re-schedules to a different time slot or date.	AVF.2.3.2
2.3.3	Go to MF.2.4	MF.2.4

	EXCEPTIONS or ERROR Flow Description	
	EF.2.3.2.2	
	Error flow at AVF.2.3.2.2: User does not re-schedule the existing appointment	
Step	**Branching Action**	**Alternate**
2.3.2.2.1	User does not override the existing appointment.	
2.3.2.2.2	User cancels the new appointment	
2.3.2.2.3	End Use Case 2.	

	ALTERNATIVE or VARIATION Flow Description	
	AVF.2.3.2	
	Alternative flow at EF.2.3.2: User re-schedules existing appointment.	
Step	**Branching Action**	**Alternate**
2.3.2.1	System asks user if they would like to override existing appointment.	
2.3.2.2	User agrees and updates the time slot and date of the existing appointment.	EF.2.3.2.2
2.3.2.3	Go to EF.1.3.3	EF.2.3.3

Non-functional Requirements for each use Case can be added in the Table below. Non-functional requirements, management issues and decisions required to be made, can be identified in the following table.

From the table below, choose a limited number of appropriate non-functional requirements relevant to the Use Case.

Non Functional Requirements can be categorised as
- Product related
- Organisation related, process and approaches set by the company
- External, imposed by outside bodies

RELATED INFORMATION	Use Case: 2	Using Scheduler System
Priority:	High	

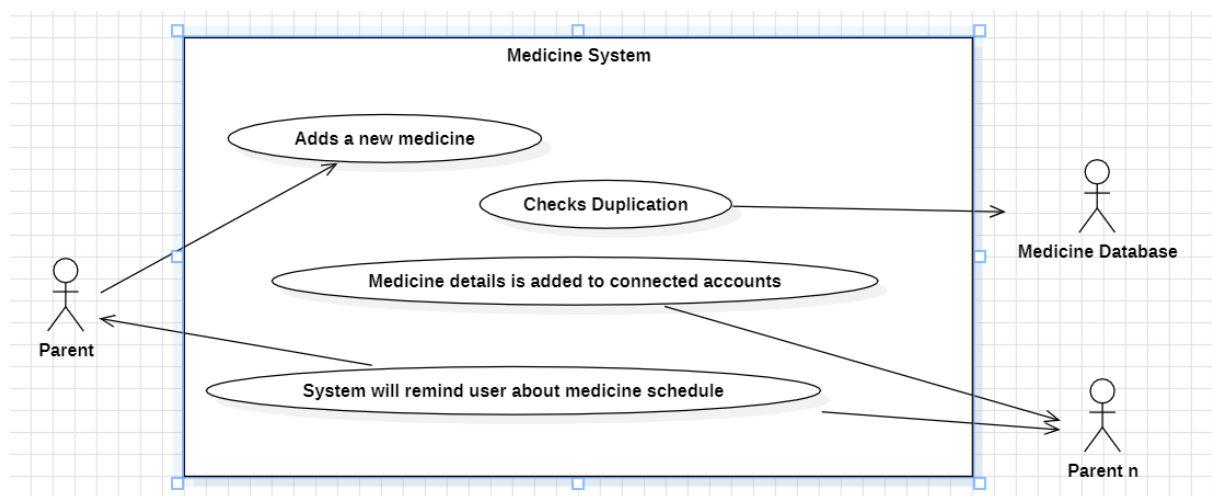| Product: Performance	Response time is essential. System should update all relevant parties about the new appointment that has been created. System should sync to cloud's calendar.
Product: Reliability	Shared acorss devices for ease of portability.
Organisation: Delivery	Electronic notifications on connected devices are set to remind user about upcoming appointments.

Use Case 3: Medicine System

Figure 2: Use Case Diagram of Medicine System (StarUML)

USE CASE	3	Using Medicine System
Description of Goal in Context	User adds a medicine profile of the medicine into the the child's profile.	
Preconditions	To access medicine system, a profile for the child must be added.	
Post Conditions, Success End Condition	After using Medicine System, user has succesfully added a medicine profile in the child's life passbook. Users connected to the child's profile will be updated and will have access to this information.	
Post Conditions, Failed End Condition	Medicine information will not be saved	
Actors	Parent/Guardian	
Trigger	User clicks a new medicine profile in Life Passbook.	

DESCRIPTION	The user creates a medicine profile which includes name, description, quantity, and duration. User can add reminders when the medication should be taken and a tracker is created to record if the medication has been taken or not.

Main Flow (MF)		
Step	Action	Alternate
3.1	User clicks (+) to add a medicine profile.	
3.2	User inputs medication details: name, description, quantity and duration.	
3.3	System will check if medicine profile already exists.	EF.3.3
3.4	System creates a new profile.	
3.5	System requests user to set a reminder for the medication.	
3.6	User inputs time and duration.	
3.7	System will provide reminders of the medication and notify user at the provided time and date.	
3.8	User marks medication has been taken.	EF.3.8
3.9	System updates medication tracker.	
3.10	End use case.	
n...	<...>	

| EXCEPTIONS or ERROR Flow Description

EF.3.3

Error flow at MF.3.3: Medication already exists. |
|---|

Step	Branching Action	Alternate
3.3.1	System notifies user about existing medication profile.	
3.3.2	User opens existing medication profile.	
3.3.3	User updates existing medication's quantity and duration.	
3.3.4	End Use Case 3	

EXCEPTIONS or ERROR Flow Description

EF.3.8		
Error flow at EF.3.8: User re-schedules reminder		
Step	Branching Action	Alternate
3.8.1	System alerts user about the medication due	
3.8.2	User re-schedules the medication time/date.	
3.8.3	Got MF.3.7	MF.3.7

Non-functional Requirements for each use Case can be added in the Table below. Non-functional requirements, management issues and decisions required to be made, can be identified in the following table.

From the table below, **choose a limited number of appropriate non-functional requirements** relevant to the Use Case.

Non Functional Requirements can be categorised as
- Product related
- Organisation related, process and approaches set by the company
- External, imposed by outside bodies

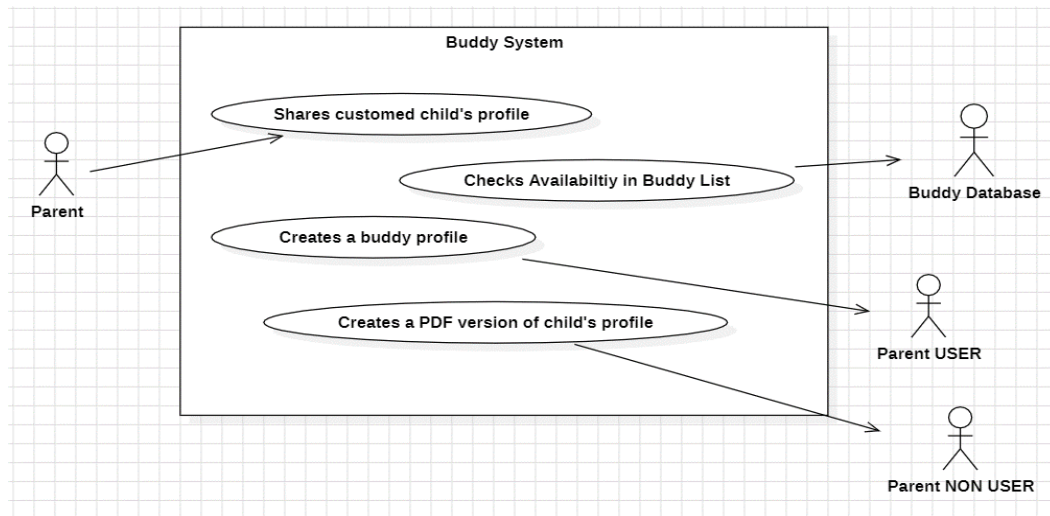| RELATED INFORMATION	Use Case: 3	Using Medicine System
Priority:	High	
Organisation: Delivery	Electronic notifications on connected devices are set to remind user about upcoming medication to be taken.	
Frequency	Based on how the user set the medication, may have 2 reminders if medication is to be taken twice a day.	
OPEN ISSUES	*What happens if two of the same medication is taken? If users can check this, how does the parents know if medication has been given?*	

Use Case 4: Playdate System

Figure 3: Use Case Diagram of Playdate System (StarUML)

USE CASE	4	Using Playdate System
Description of Goal in Context	User shares a customed profile of their child to relavent party.	
Preconditions	To access Playdate system, a profile for the child must be added. User should have an existing relationship with the child to share information via Agape directly.	
Post Conditions, Success End Condition	After using Playdate System, user has succesfully shared a customed profile of their child to the relevant party.	
Post Conditions, Failed End Condition	Child's profile will not be shared.	
Actors	Parent/Guardian	
Trigger	User clicks the share feature on their child's profile.	
DESCRIPTION	The user shares their child's customised profile to other parties. User is able to control what information is to be shared. User can share directly with an Agape user or system will allow the option to convert profile into a PDF.	

Main Flow (MF)		
Step	Action	Alternate
4.1	User clicks share child's profile.	

4.2	System will ask what information user would like to share.	
4.3	User marks which information they would like to share.	
4.4	System will display the information that will be shared.	
4.5	User confirms the information.	
4.6	System asks which user the profile will be shared with.	AVF.4.6
4.7	User chooses the user which the information will be sent to.	EF.4.7
4.8	System will prompt the receiving user if they would like to add the profile in their Buddy System.	
4.9	System sends the information and saves to the receiving user's account.	
4.10	System notifies user that the profile has been shared with the chosen user.	
4.11	End use case.	
n...	<...>	

EXCEPTIONS or ERROR Flow Description

EF.4.7

Error flow at MF.4.7: Agape User does not have an existing relationship with the child.

Step	Branching Action	Alternate
4.3.1	Users add each other.	
4.3.2	System creates a relationship of the added user to the child.	
4.3.3	Go to MF.4.7	MF.4.7

ALTERNATIVE or VARIATION Flow Description

AVF.4.6

Alternative flow at MF.4.6: User wants to share to a party that is not an Agape User

Step	Branching Action	Alternate
4.6.1	User clicks "Other Option"	
4.6.2	System requests user to choose what media they would like to send the customed profile	

4.6.3	User chooses media (SMS/Email/Print/etc.)	
4.6.4	System converts profile into a PDF	
4.6.5	User can save or send the PDF directly to the chosen party	
4.6.6	End Use Case 4	

Non-functional Requirements for each use Case can be added in the Table below. Non-functional requirements, management issues and decisions required to be made, can be identified in the following table.

From the table below, **choose a limited number of appropriate non-functional requirements** relevant to the Use Case.

Non Functional Requirements can be categorised as
- Product related
- Organisation related, process and approaches set by the company
- External, imposed by outside bodies

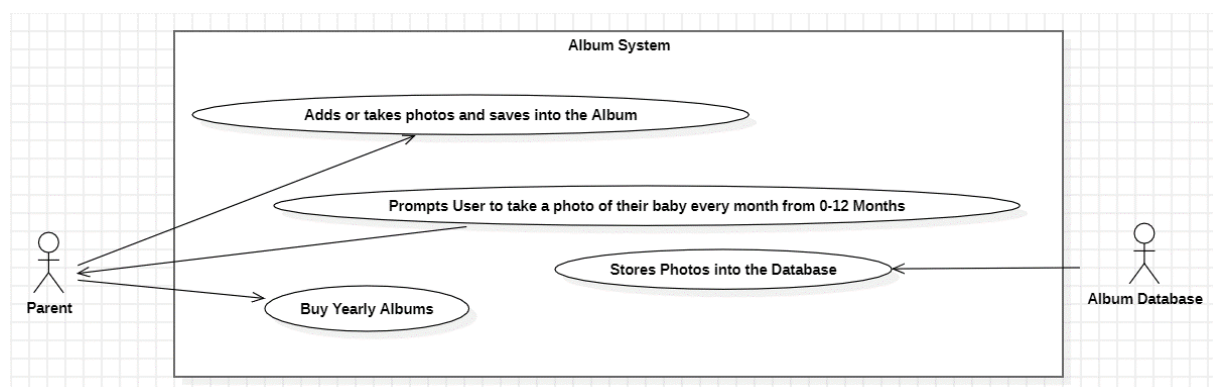| RELATED INFORMATION	Use Case: 4	Using Playdate System
Priority:	Mid Priority	
Product: Performance	Response time is essential. Sharing profiles should take less than 2 minutes to connect.	
Product: Reliability	Profile is able to be converted to PDF if profile is to be shared with a non-user.	
Organisation: Delivery	Important notes such as allergies will be highlighted in the profile provided – prompts are based on level of importance.	

Use Case 5: Album System

Figure 4: Use Case Diagram of Album System (StarUML)

USE CASE	5	Using Album System
Description of Goal in Context	User is able to buy a physical album from Agape	
Preconditions	To access Album system, a profile for the child must be added and payment details must be verified.	
Post Conditions, Success End Condition	After using Album System, user has bought an album product from Agape.	
Post Conditions, Failed End Condition	Inability to buy album if payment requirements are invalid.	
Actors	Parent/Guardian	
Trigger	System will prompt a reminder to user, to buy an album or SD card every year of the photos taken/saved in the cloud of that year. User can also directly access this service in Album Feature.	
DESCRIPTION	User can order a physical album to be sent to their home of selected photos from the photo album. User can also opt to add or just buy an SD card for the (unselected) photos.	

Main Flow (MF)		
Step	Action	Alternate
5.1	User buys a physical album(s).	
5.2	System requests user to choose photos from the cloud album.	
5.3	User selects photos to be included.	
5.4	System asks users what type of album style they would like.	
5.5	User can choose from the variety of album styles and add a message.	
5.6	System shows user a sample of how the album will look like.	
5.7	User confirms the choice of album.	EF.5.7
5.8	System request for user details: Name, address, email and other information.	
5.9	User inputs requested information.	
5.10	System will display total amount for the album.	

5.11	User provides payment details.	
5.12	System verifies payment details.	EF.5.12
5.13	System creates an order for the album.	
5.14	System sends user an email of reciept.	
5.15	End use case.	
n...	<...>	

EXCEPTIONS or ERROR Flow Description
EF.5.7
Error flow at MF.5.7: User is not happy with sample Album

Step	Branching Action	Alternate
5.3.1	System reverts to edit phase	
5.3.2	User updates their preference in Album style	
5.3.3	Go to MF.5.6	MF.5.6

EXCEPTIONS or ERROR Flow Description
EF.5.12
Error flow at EF.5.12: Payment Verification Failed

Step	Branching Action	Alternate
5.12.1	System notifies user that payment verification failed.	
5.12.2	User updates payment details.	
5.12.3	Go to MF.5.12	M.5.12

Non-functional Requirements for each use Case can be added in the Table below. Non-functional requirements, management issues and decisions required to be made, can be identified in the following table.

From the table below, choose a limited number of appropriate non-functional requirements relevant to the Use Case.

Non Functional Requirements can be categorised as
- Product related

- Organisation related, process and approaches set by the company
- External, imposed by outside bodies

RELATED INFORMATION	Use Case: 5	Using Album System
Priority:	Mid Priority	
Product: Performance	Response time is essential. Payment verification should take less than a minute.	
Product: Reliability	System reminds user about yearly album a week prior to child's birthday.	
Organisation: Delivery	Physical album/SD Card is to be sent to the provided address of the order.	
OPEN ISSUES	What happens when application has no more storage to save photos?	

External Database or System

In particular identify at least one actor that is an external Database or system which will interact with your system.

For each use case, an external database will interact with my system. They are…

Scheduler Database	This role is responsible for securely storing and managing data related to appointments, trackers, and other scheduling information within the application.
Medicine Database	Responsible for maintaining a secure database containing information about the child's medicinal history - ensuring the child's health details are easily accessible to parents and can share this to medical professionals.
Buddy Database	Tasked with managing the database that stores playdate information, including details about the child's friends and their parents. This feature enhances social connections, facilitates playdate planning and ensure the safety of the child.
Album Database	Manages the storage of photos, videos, and information related to album purchases.
Parent Database (Non-User, User(n))	Parents who may not actively use the app or not a user at all, can still benefit from the application. Primary users can share specific information with them. This feature promotes inclusivity and effective communication between app users and those who are not users.

Further Analysis

Complete a third iteration of use case modelling. Identify two appropriate use case for further analysis.
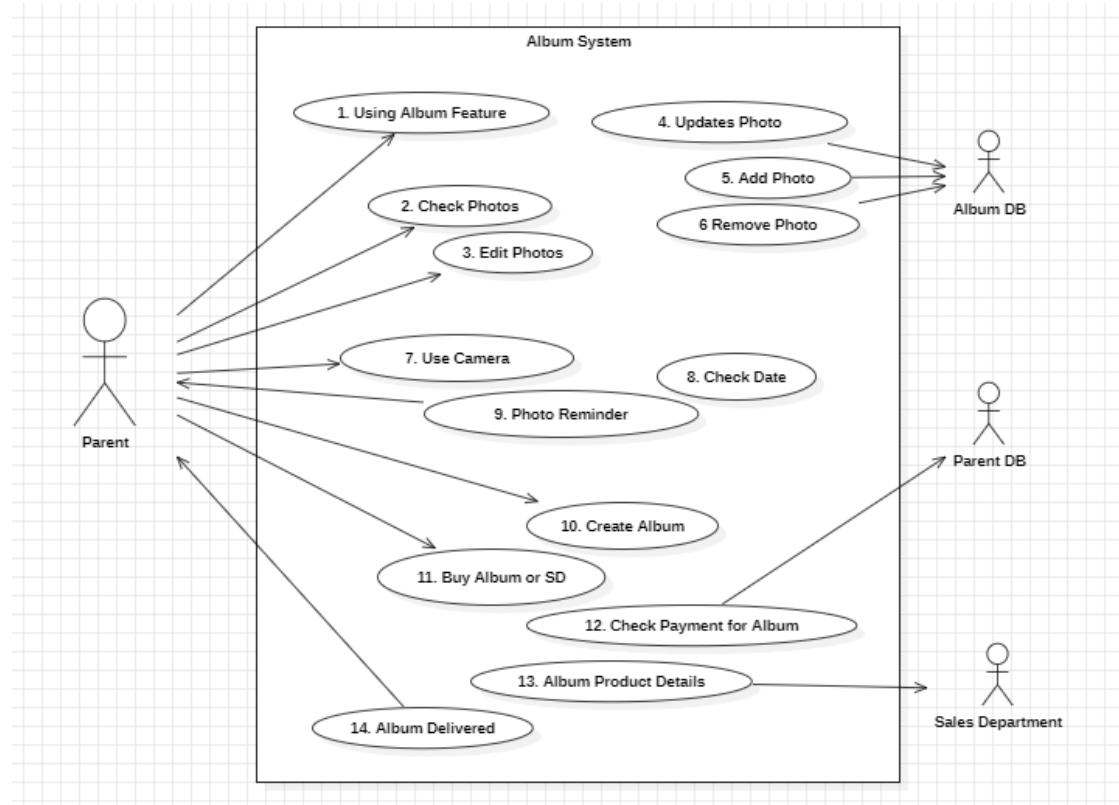
Album System

Figure 5: Use Case Diagram of Album System (StarUML) Further Analysis

USE CASE	5	Using Album System
Description of Goal in Context	The user (parent) wants to capture, organise and store their child's photos of key moments using the Album System. This feature allows users to create a digital album of their child's development milestones and have the option to order a physical copy delivered directly to their house.	
Preconditions	To access Album system, a profile for the child must be added and has accessed the Album & Camera feature of the Agape.	
Post Conditions, Success End Condition	Parent has successfully added, organised and updated the photo album of the key moments of their child. An album will be available for purchase including recommended photos or photos personally chosen by the parent.	

Post Conditions, Failed End Condition	The parent experiences errors while attempting to use the Almbun feature dur to technical issues or the app crashes. The parent may also fail to buy an album if payment requirements are invalid.
Actors	Parent/Guardian
Trigger	User wants to capture a key moment of their child such as first steps, birthday or a memorable outing. They are able to access the camera of the app which includes various filters or built-in editing tools. User may also want to upload or download photos into/from the album database.
DESCRIPTION	The parent is able to document their child's growth via the Album System. This feature provides tools to capture photos, organise them and add descriptions or notes. The app also includes built-in editing tools to create a customised album.

Main Flow (MF)		
Step	Action	Alternate
5.1	User access the Album & Camera feature of the app.	
5.2	The system displays the main interface of this feature which includes, a photo feed, options to add/upload/download photos as well as creating an album, and a camera option.	
5.3	The user captures a photo via the camera app.	EF.5.3
5.4	System provides option to edit photo and add a description.	
5.5	User writes a decription of the photo.	
5.6	System adds the photo into the Album Database.	
5.7	User creates an Album and choses photos to be included.	EF.5.7
5.8	System saves the album into the parent's profile to be access for editing or to purchase at a future date.	
5.9	User decides to buy the album.	
5.10	System validates payments and delivery destination.	EF.5.10
5.11	User proivdes delivery information and payments.	
5.12	System shares the product information to sales department to be readied for delivery.	
5.13	User recieves the customised Album.	
5.14	User oragnises photos in specific categories as well as editing photos via the app tools.	

5.15	System updates changes to photos into the Album Database.	
5.16	End Use Case 5.	
n…	<…>	

EXCEPTIONS or ERROR Flow Description		
EF.5.3		
Error flow at MF.5.7: No Camera Permissions		
If the user access the app's camera to capture a photo without granting camera permissions, the app displays an error message and prompts the user to enable camera access from settings.		

Step	Branching Action	Alternate
5.3.1	System prompts an error message requesting for the user to grant camera permission from settings.	
5.3.2	User allows camera permission from Settings and returns to the app's Camera.	
5.3.3	User takes a photo.	
5.3.4	Go to MF.5.4	MF.5.4

EXCEPTIONS or ERROR Flow Description		
EF.5.10		
Error flow at EF.5.12: Payment Verification Failed		

Step	Branching Action	Alternate
5.10.1	System notifies user that payment verification failed.	
5.10.2	User updates payment details.	
5.10.3	Go to MF.5.10	M.5.10

Non-functional Requirements for each use Case can be added in the Table below. Non-functional requirements, management issues and decisions required to be made, can be identified in the following table.

From the table below, **choose <u>a limited number</u> of appropriate non-functional requirements** relevant to the Use Case.

Non Functional Requirements can be categorised as

- Product related
- Organisation related, process and approaches set by the company

- External, imposed by outside bodies

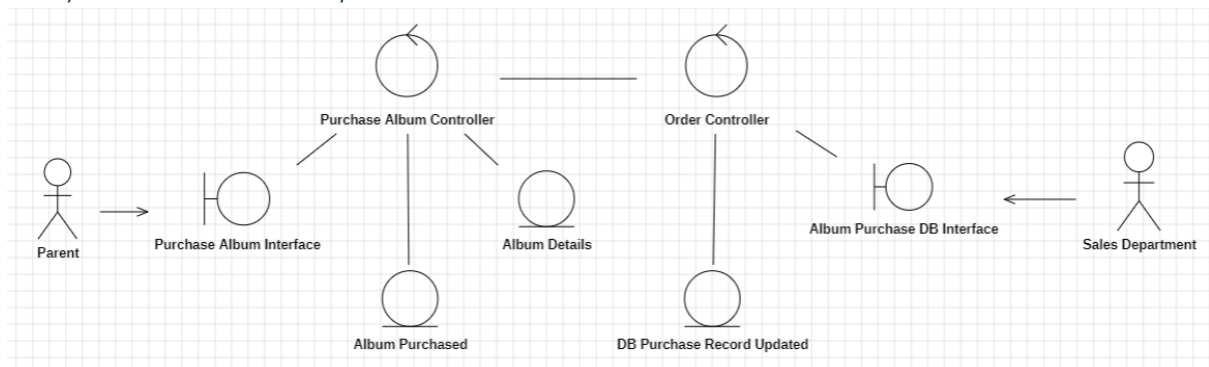| RELATED INFORMATION	Use Case: 5	Using Album System
Priority:	High	
Product: Performance	Dynamic UI and quick loading time for viewing as well as scrolling through the photo feed.	
Product: Efficiency	UI must be user-friendly, camera should be accesibile and easy to use. Photos should be organised and be able to share to the top media platforms.	
Product: Reliability	The album system must be reliable to store and retrieve photos without data loss issues.	
Organisation: Delivery	Be able to deliver album within 2-3 days business days where 2 weeks as max delay for delivery.	
Frequency	Any time user want to use to Album System, ex. multiple times a week.	
Channels to actors	Interactive usage such a photo capture, organisation and sharing.	
OPEN ISSUES	Data backup and synchronization methods to prevent photo loss in case of app crashes or technical issues.	

Analysis Class for Album System

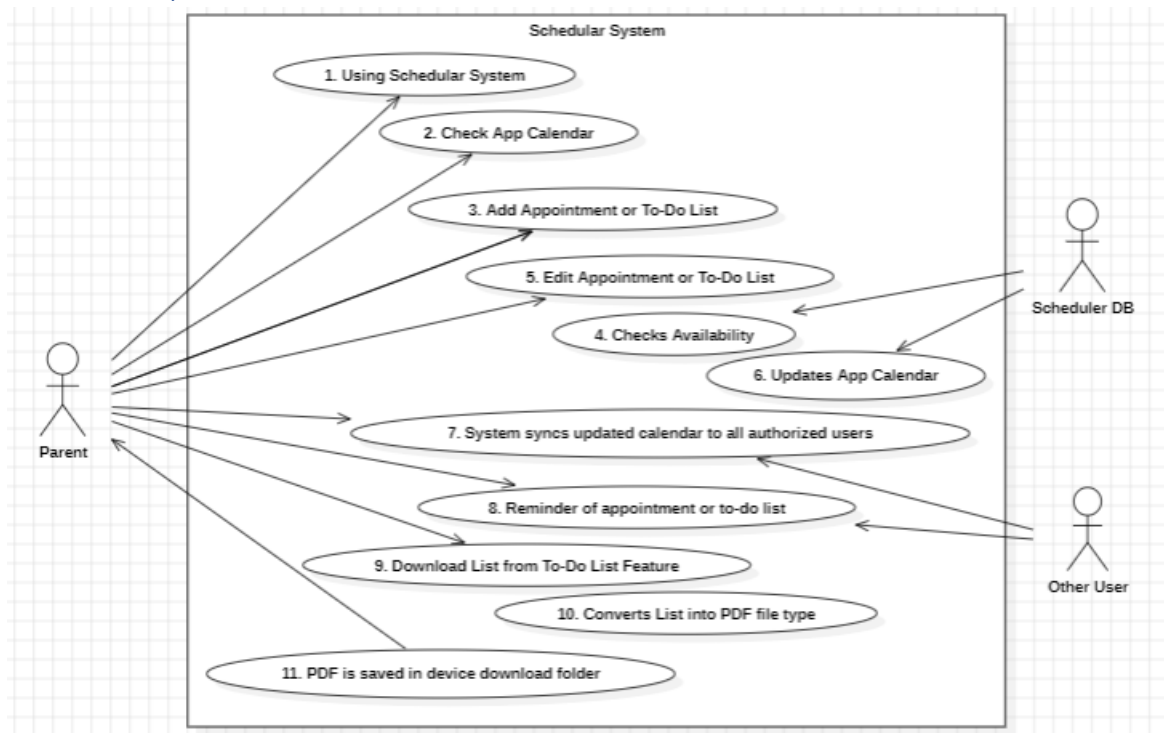*Figure 6: Analysis Class for Album System*

Schedular System

Figure 1: Use Case Diagram of Scheduler System (StarUML) Further Analysis

USE CASE	2	Using Scheduler System
Description of Goal in Context	The parent is able to manage their parenting-related activities via the Scheduler System. This feature includes scheduling appointments, setting reminders and creating a to-do list which are all saved unto the system's calendar and synced with other authorized users.	
Preconditions	To access scheduler system, a profile for the child must be added and user accessed the Scheduler System.	
Post Conditions, Success End Condition	User has succesfully adds an appointment on the calendar and other users connected to the child's profile will be updated and have access to this appointment. At the set date, the system will remind all related users about the upcoming appointments.	
Post Conditions, Failed End Condition	Unable to save appointment due to technical or network issues.	
Actors	Parent/Guardian	
Trigger	User has an appointment to add or a reminder is set off for a upcoming schedule.	

DESCRIPTION	The user creates an appointment which includes scheduling appointments, setting reminders, and creating and managing to-do lists. This will be saved in the system and synced into relevant users.

Main Flow (MF)		
Step	Action	Alternate
2.1	User access the Scheduler System.	
2.2	System displays a calendar UI which includes options to add a new appointment and the to-do list management.	
2.3	User adds an appointment.	
2.4	System request the date, time and a brief description of the appointment.	
2.5	User adds the appointment information.	
2.6	System checks availabilty on the calendar.	EF.2.6
2.7	System updates the new appointment on the app Calendar and syncs to other authorized users' calendar.	
2.8	User is able to check upcoming events on the calendar.	
2.9	User access the to-do list for groceries and adds a new item on the list.	
2.10	User saves the list on their phone.	
2.11	System converts the list into a PDF and saves unto the user's device's download folder.	
2.12	End of Use Case 2	
n...	<...>	

EXCEPTIONS or ERROR Flow Description
EF.2.6
Error flow at MF.2.6 : Timeslot and date have been booked.

Step	Branching Action	Alternate
2.6.1	System notifies user about existing appointment on the given date and time.	
2.6.2	User re-schedules to a different time slot or date.	AVF.2.6.2

2.6.3	Go to MF.2.7	MF.2.7

ALTERNATIVE or VARIATION Flow Description
AVF.2.6.2
Alternative flow at EF.2.3.2: User re-schedules existing appointment.

Step	Branching Action	Alternate
2.6.2.1	System asks user if they would like to override existing appointment.	
2.6.2.2	User agrees and updates the time slot and date of the existing appointment.	
2.6.3	Go to MF.2.7	MF.2.7

Non-functional Requirements for each use Case can be added in the Table below. Non-functional requirements, management issues and decisions required to be made, can be identified in the following table.

From the table below, **choose <u>a limited number</u> of appropriate non-functional requirements** relevant to the Use Case.

Non Functional Requirements can be categorised as
- Product related
- Organisation related, process and approaches set by the company
- External, imposed by outside bodies

RELATED INFORMATION	Use Case: 2	Using Scheduler System
Priority:	High	
Product: Efficiency	The interface should be user-friendly. Add, edit and delete should be easy to use.	
Product: Reliability	Reminders must prompt on the set date without delay. Appointment data must be secured. Operation/process must always be stable.	
Organisation: Delivery	Electronic notifications on connected devices are set to remind user about upcoming appointments.	
Frequency	User may interact with this feature multiple times a day.	
Channels to actors	Real-time data updates.	

OPEN ISSUES	Data synchronisation methods to ensure consistency across multiple devices and authorized users.

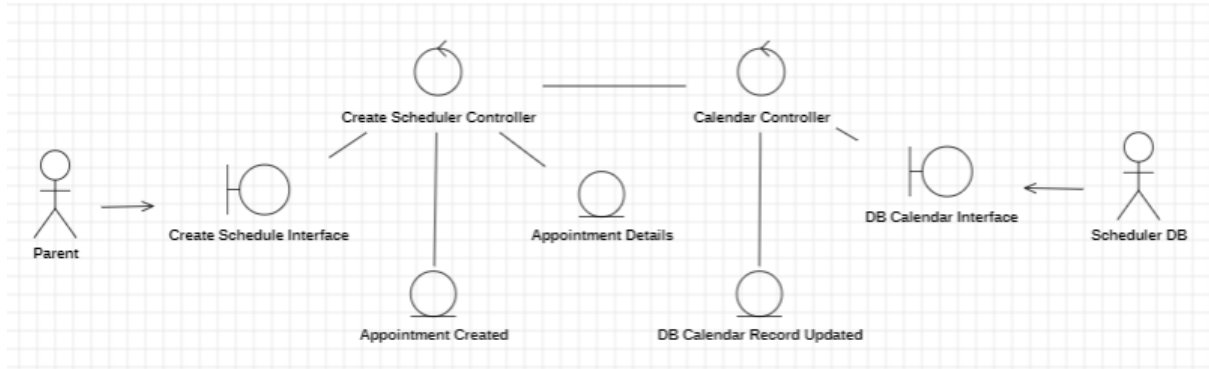## Analysis Class for Scheduler System

Figure 7: Analysis Class for Scheduler System

Sequence Diagram

Using the analysis classes, from week 5, create one communication/collaboration diagram, in the form of a sequence diagram, to analyse the main flows for the 2 use cases under consideration.

Then identify the design classes from the analysis classes. Draw a sequence diagram based on the design classes.
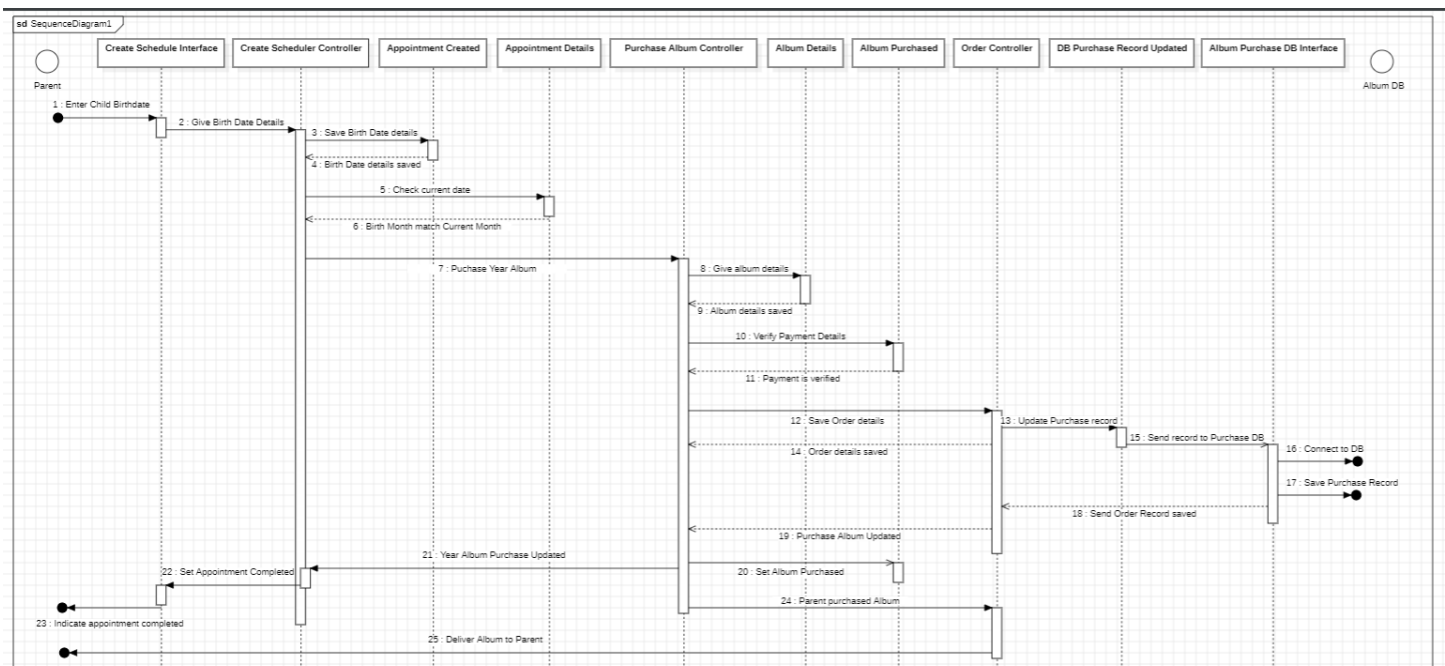
Improper Sequence Diagram

Figure 8: Improper Sequence Diagram of Album and Scheduler Use Cases
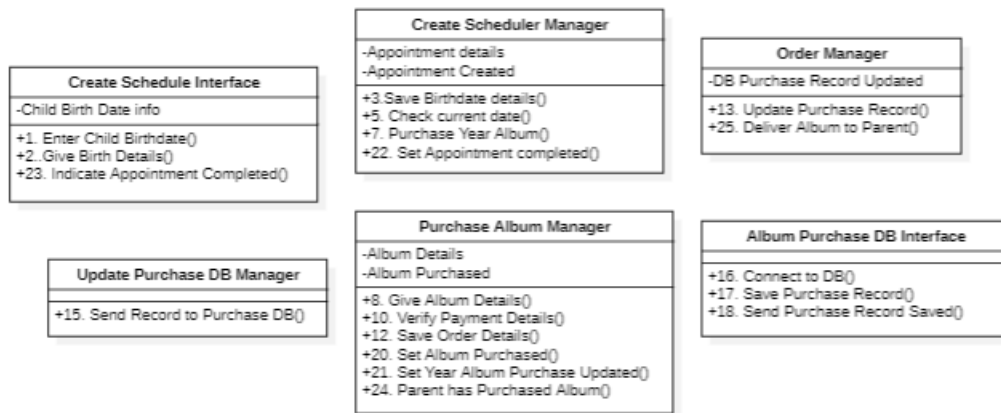
Design Classes

Figure 9: Design Classes of Album and Scheduler Use Cases
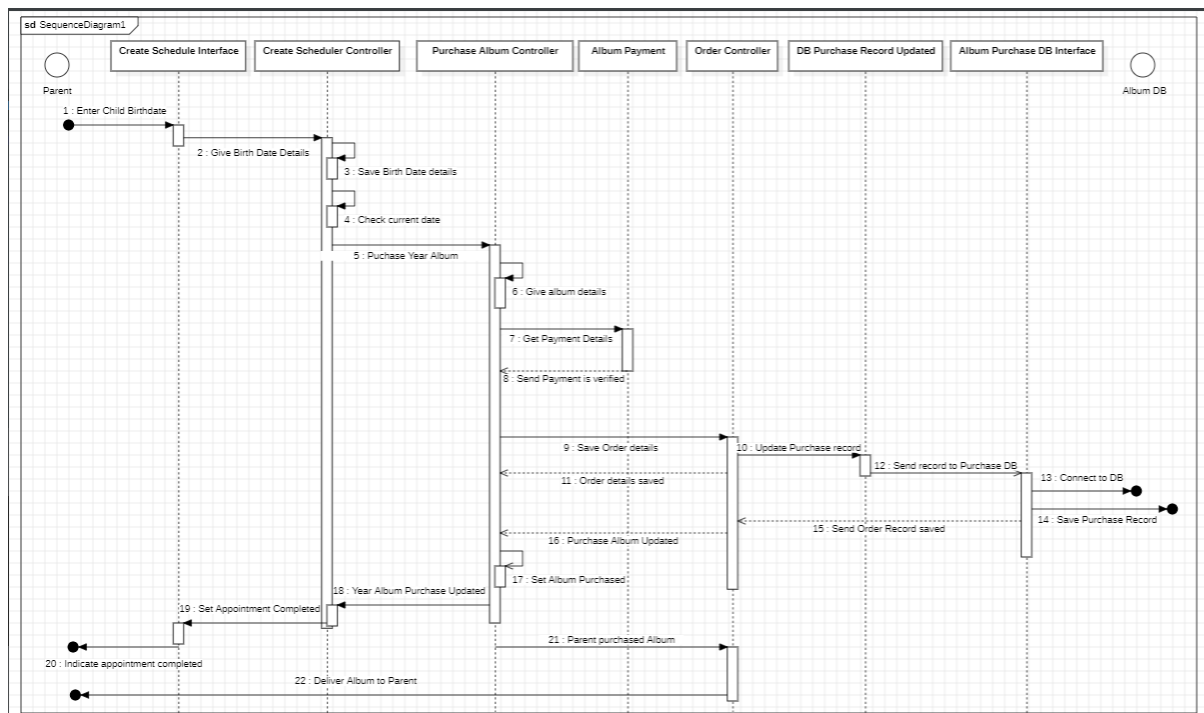
Proper Sequence Diagram

Figure 10: Proper Sequence Diagram of Album and Scheduler Use Cases