

**COMP1531**

**Project Backend [Team Echo]**

*5.3. Planning for the next problems to solve*

*Brendan Ye, Darrell Mounarath, Kellen Liew, Nikki Yao, Winston Lin*

# Table of Contents

## [1 | Requirements](#)

### [1.1 | Elicitation](#)

### [1.2 | Analysis & Specification - Use Cases](#)

### [1.3 | Validation](#)

## [2 | Design](#)

### [2.1 | Interface Design](#)

### [2.2 | Conceptual Modelling \(State\)](#)

## 1 | Requirements

### 1.1 | Elicitation

#### Target Users

*Demographic: university students using Dreams for team situations*

NAME	EMAIL ADDRESS
Christian	christian.huang@gmail.com
Phillip	phillip.leung@gmail.com
Alison	alison.zhong@gmail.com

#### Objective

To understand what *problems* the target demographic might have with teamwork-driven communication tools that are currently unsolved by **Dreams**.

#### Questions

*\*Prior to the interviews, the users were briefed on the current functionality of Dreams and what type of platform it is.*

1. What communication tools do you currently use for teamwork?
2. What purpose do they use the platform for?
3. What do you like about Microsoft Teams/Discord for messaging in team situations?
4. What would you wish to be able to do on the platform that you cannot do right now?
5. Given what Dreams has currently implemented, what do you think would be the next most important thing implemented to ensure a good user experience in team communication?

#### User 1 (Christian Huang)

1. What communication tools do you currently use for teamwork, and any specifically alike Dreams?

*Currently, Facebook Messenger, Discord, sometimes Slack, and Microsoft Teams. If asking for a platform alike to Dreams; probably Microsoft Teams.*

2. What purpose do you use [Microsoft Teams] for?

*I use Microsoft Teams for university course work as they provide it as a requirement to stay updated with the class, and it is used heavily for work.*

3. What do you like about [Microsoft Teams] for messaging in team situations?

*At work, I think it is pretty cool that Teams has the project cards function, but if I were to focus on university team situations, I like that Teams has the ribbon on top of the chats to easily access files.*

4. What do you wish to be able to do on the platform that you cannot do right now?

*I wish that the process of working collaboratively can be smoother - be it organising times for meetups or having a section to jot down important notes.*

5. Given the current functionality of Dreams, what do you think would be the next most important thing implemented to ensure a good user experience in team communication?

*One should definitely be able to send files in the channels as working in teams calls for collaborative working on digital documents.*

*Another would be a collaborative calendar as it allows for a more holistic view of ongoing projects, or just to see your members' schedules.*

#### User 2 (Phillip Leung)

1. What communication tools do you currently use for teamwork, and any specifically alike Dreams?

*I use Microsoft Teams as my university course requires it. Other communication tools would include [Facebook] Messenger, sometimes Slack, and Discord.*

2. What purpose do you use [Microsoft Teams] for?

*University society work and it is compulsory for one of my courses.*

3. What do you like about [Microsoft Teams] for messaging in team situations?

*I like that it:*

- a. Serves its purpose,*
- b. I can seeing everyone react to stuff,*
- c. It is quite integrated with workflow, i.e. microsoft products.*

4. What do you wish to be able to do on the platform that you cannot do right now?

*Cross calendars - I think that is pretty essential to a smooth workflow. To organise meetups well and also to pinpoint key dates.*

5. Given the current functionality of Dreams, what do you think would be the next most important thing implemented to ensure a good user experience in team communication?

*Some things that I would like:*

- a. *Sorting public/private chats*
- b. *Pin messages*
- c. *Calendar time meetings where everyone can view the calendar*

### User 3 (Alison Zhong)

1. What communication tools do you currently use for teamwork, and any specifically alike Dreams?

*[Facebook] Messenger and Discord heavily for society work. MS teams for a course I am doing.*

2. What purpose do you use [Microsoft Teams] for?

*Required for a course I am doing.*

3. What do you like about [Microsoft Teams] for messaging in team situations?

*I like that everything is integrated and easily accessible. I.e. I really dig the top ribbon bar of chat windows where I can just pin apps relevant to the chat.*

4. What do you wish to be able to do on the platform that you cannot do right now?

*At the beginning of my university course, I was confused about who the tutor was in the team. If teams could create a status or category to tag people with [like Discord], that would be great.*

*And also cross calendar checks with the members of the channel - that would ease the process of finding a suitable time.*

5. Given the current functionality of Dreams, what do you think would be the next most important thing implemented to ensure a good user experience in team communication?

*Some things that I would like:*

- a. *Creating tags/categories to sort members out*
- b. *Calendar time meetings where everyone can view the calendar*

### Proposed Solution

From interviewing peers in the university network and analysing trends in their responses, it can be determined that scheduling meetings and understanding each other's availability are essential features in team situations. Whilst messaging is sufficient to update each other on certain topics, in university scenarios like group projects, collaborative work is deemed more effective when all

members unite to focus on the project at hand. The target users mentioned the necessity of voice and video conferencing, and in-person meet-ups; prior to these meetings, it is important to find the most appropriate time to have it.

Dreams should aim to incorporate a calendar feature in channels and dms which allows users to set an appropriate time for meetings. Users will be able to write down their commitments in their personal calendar. Members of a channel or dm will be able to merge their calendars and determine a convenient time to meet.

## 1.2 | Analysis & Specification - Use Cases

### User Scenario 1

#### *User Story*

- **As a** university student
- **I can** see my peers' schedules
- **so that** I can schedule the appropriate time for meetings.

#### *User Acceptance Criteria*

**Scenario:** Finding the most appropriate time for a meeting

**Given:** The user has navigated to their group channel page

**When:** The user selected the calendar feature

**And:** The user clicked the channel members calendar

**Then:** The system shows the user the combined calendar of members in the channel

### User Scenario 2

#### *User Story*

- **As a** university student
- **I can** note important deadlines or commitments in my calendar
- **so that** I can manage my time appropriately and feel more in control of my schedule.

#### *User Acceptance Criteria*

**Scenario:** Note important events in the calendar

**Given:** The user has navigated to their home page

**When:** The user hovered over the calendar icon

**And:** Found the date to note an important event

**When:** The user clicked the date to add an important event

**Then:** The system enables the user to insert and display the event in the calendar

## Use Case 1

### *Team Calendar*

- **Use Case:** View team calendar
- **Goal in Context:** Users need to view their channel members' calendars to more conveniently determine a time to meet
- **Scope:** Network application
- **Level:** Subfunction
- **Preconditions:** The user has an account with UNSW Dreams and is in a channel(s)/dm(s)
- **Success End Condition:** The user successfully determines a meeting time/s within their channel/dm faster than without a team calendar
- **Failed End Condition:** The user cannot determine an appropriate time as other user's have not input their commitments and availability
- **Primary Actor:** User
- **Trigger:** User logs into UNSW Dreams

### *Main Success Scenario*

Step 1. User clicks channel calendar

Step 2. User observes the merged key events of members in the channel

Step 3. User determines a time to meet

Step 4. User creates a key event in the team calendar

Step 5. The new key event is appended to all the channel members' personal calendars.

## Use Case 2

### *Personal calendar*

- **Use Case:** View and add events into personal calendar
- **Goal in Context:** Users need to view their personal calendar to manage their time appropriately
- **Scope:** Network application
- **Level:** Subfunction

- **Preconditions:** The user has an account with UNSW Dreams
- **Success End Condition:** The user views their personal calendar for upcoming events
- **Failed End Condition:** The user has nothing noted in the calendar
- **Primary Actor:** User
- **Trigger:** User logs into UNSW Dreams

### *Main Success Scenario*

Step 1. User clicks personal calendar

Step 2. User locates a date to add a key event

Step 3. User clicks on the date to create a key event

Step 4. User fills in the details of the event, i.e. title, detail, date, time start, time end, participants

Step 5. User clicks 'Create event'

Step 6. The newly created event is added to the user's personal calendar

## 1.3 | Validation

### Interviewee 1 (Christian Huang)

#### Comments

*It seems pretty useful. I think drawing similarities to when2meet (an existing collaborative event tool) will be good - showing the interactions of different peoples' availability to help schedule meetings. And yes, I'd hope that if there was a calendar function I'd be able to open it and add my own events.*

*I guess adding meeting links or agendas or link to minutes or smth would be cool as well.*

### Interviewee 2 (Phillip Leung)

#### Comments

*The use cases definitely represent how I would use those functionalities.*

*It would definitely help with engagement as your workflow is much faster as everything's in one place as well.*

*For improving team coordination, it[the proposed features] makes it easier to schedule potential dates and times as you can see available slots.*



Interviewee 3 (Alison Zhong)

Comments

*This is good! As I have many meetings - be it work or society or group projects, the calendar will be indicative of the most suitable time for our meet-ups. And also in situations where someone doesn't turn up, hopefully we'll be able to see their commitments - it allows for more transparency in group settings.*

*I also like that I will be able to pin things in a personal calendar on the platform. Definitely allows for more integration - and just having university commitments in one place.*

## 2 | Design

### Data Types

Variable Name	Type
Has suffix <b>id</b>	integer
Named exactly <b>token</b>	string
Named exactly <b>title</b>	string
Named exactly <b>detail</b>	string
Named exactly <b>date</b>	integer
Has prefix <b>time_</b>	integer (unix timestamp)
Has suffix <b>start</b>	integer
Has suffix <b>end</b>	integer
(outputs only) Named exactly <b>event</b>	Dictionary containing title, detail, date, time_start, time_end
(outputs only) Named exactly <b>events</b>	List of dictionaries, where each dictionary contains types { title, detail, date, time_start, time_end }

### 2.1 | Interface Design

Name & Description	HTTP Method	Data Types	Exceptions
<b>calendar/create</b>  A valid token creates a key event in their calendar by providing the title, detail, date, time_start, time_end.	POST	<b>Parameters:</b> (token, title, detail, date, time_start, time_end )  <b>Return Types:</b> { event_id }	<b>Input Error</b> when any of: <ul style="list-style-type: none"><li>• Title is more than 500 characters long</li><li>• Detail is more than 1000 characters long</li><li>• Time_start and time_end is not within 0000 and 2400</li><li>• Time_end is earlier than time_start</li></ul> <b>Access Error</b> when any of:

			<ul style="list-style-type: none"> <li>Token is invalid</li> </ul>
<b>calendar/events</b>  Provide a list of events (and their associated details) between start date and end date of the authorised user's calendar.	GET	<b>Parameters:</b> (token, start, end)  <b>Return Types:</b> { events }	<b>Input Error</b> when any of: <ul style="list-style-type: none"> <li>Start and end span more than 365 days</li> <li>End is earlier than start</li> </ul> <b>Access Error</b> when any of: <ul style="list-style-type: none"> <li>Token is invalid</li> </ul>
<b>calendar/details</b>  Given an event_id in the authorised user's calendar, provide basic details about the event.	GET	<b>Parameters:</b> (token, event_id)  <b>Return Types:</b> { event }	<b>Input Error</b> when any of: <ul style="list-style-type: none"> <li>Event_id is invalid</li> <li>Event_id is already removed</li> </ul> <b>Access Error</b> when any of: <ul style="list-style-type: none"> <li>Token is invalid</li> <li>Token is not a part of the event_id</li> </ul>
<b>calendar/edit</b>  Given an event_id, update its field with either or a combination of the following: <ul style="list-style-type: none"> <li>title</li> <li>detail</li> <li>date</li> <li>time_start</li> <li>time_end</li> </ul> When a key is not being edited, the input will be '-1.'	PUT	<b>Parameters:</b> (token, event_id, title, detail, date, time_start, time_end)  <b>Return Types:</b> { }	<b>Input Error</b> when any of: <ul style="list-style-type: none"> <li>Event_id is invalid</li> <li>Event_id is removed</li> <li>Title is more than 500 characters long</li> <li>Detail is more than 1000 characters long</li> <li>Time_start and time_end is not within 0000 and 2400</li> <li>Time_end is earlier than time_start</li> </ul> <b>Access Error</b> when any of: <ul style="list-style-type: none"> <li>Token is invalid</li> <li>Token is not a part of the event_id</li> </ul>

<b>calendar/remove</b>  Given an event_id for an event, remove from the authorised user's calendar.	DELETE	<b>Parameters:</b> (token, event_id)  <b>Return Types:</b> {}	<b>Input Error</b> when any of: <ul style="list-style-type: none"> <li>• Event_id is invalid</li> <li>• Event_id is already removed</li> </ul> <b>Access Error</b> when any of: <ul style="list-style-type: none"> <li>• Token is invalid</li> <li>• Token is not a part of the event_id</li> </ul>
<b>team/calendar/create</b>  A valid token creates a key event in the given channel_id/dm_id calendar by providing the title, detail, date, time_start, time_end.  This event will also be added to the personal calendars of the members in channel/dm.	POST	<b>Parameters:</b> (token, channel_id, dm_id, title, detail, date, time_start, time_end)  <b>Return Types:</b> { event_id }	<b>Input Error</b> when any of: <ul style="list-style-type: none"> <li>• Channel_id is invalid</li> <li>• Dm_id is invalid</li> <li>• Neither channel_id or dm_id has the input '-1'</li> <li>• Title is more than 500 characters long</li> <li>• Detail is more than 1000 characters long</li> <li>• Time_start and time_end is not within 0000 and 2400</li> <li>• Time_end is earlier than time_start</li> </ul> <b>Access Error</b> when any of: <ul style="list-style-type: none"> <li>• Token is invalid</li> <li>• Token is not a part of the given channel_id or dm_id</li> </ul>

<p><b>team/calendar/events</b></p> <p>Provide a list of events (and their associated details) between start date and end date of the channel/dm calendar.</p> <p>When viewing a channel_id calendar, the input for dm_id is '-1' and vice versa.</p> <p>Personal events of members in the channel_id/dm_id will be shown but limited to title, date and time.</p>	GET	<p><b>Parameters:</b> (token, channel_id, dm_id, start, end)</p> <p><b>Return Types:</b> { events }</p>	<p><b>Input Error</b> when any of:</p> <ul style="list-style-type: none"> <li>Channel_id is invalid</li> <li>Dm_id is invalid</li> <li>Neither channel_id or dm_id has the input '-1'</li> <li>Start and end span more than 365 days</li> <li>End is earlier than start</li> </ul> <p><b>Access Error</b> when any of:</p> <ul style="list-style-type: none"> <li>Token is invalid</li> <li>Token is not a part of the given channel_id or dm_id</li> </ul>
<p><b>team/calendar/details</b></p> <p>Given an event_id in the channel/dm calendar, provide basic details about the event.</p>	GET	<p><b>Parameters:</b> (token, channel_id, dm_id, event_id)</p> <p><b>Return Types:</b> { event }</p>	<p><b>Input Error</b> when any of:</p> <ul style="list-style-type: none"> <li>Channel_id is invalid</li> <li>Dm_id is invalid</li> <li>Neither channel_id or dm_id has the input '-1'</li> <li>Event_id is invalid</li> <li>Event_id is already removed</li> </ul> <p><b>Access Error</b> when any of:</p> <ul style="list-style-type: none"> <li>Token is invalid</li> <li>Token is not a part of the given channel_id or dm_id</li> <li>Token is not a participant of the event_id</li> </ul>

<b>team/calendar/edit</b>  Given an event_id, update its field with the either or a combination of the following: <ul style="list-style-type: none"> <li>• title,</li> <li>• detail,</li> <li>• date,</li> <li>• time_start,</li> <li>• time_end.</li> </ul>	PUT	<b>Parameters:</b> (token, event_id, title, detail, date, time_start, time_end)  <b>Return Types:</b> {}	<b>Input Error</b> when any of: <ul style="list-style-type: none"> <li>• Event_id is invalid</li> <li>• Event_id is removed</li> <li>• Title is more than 500 characters long</li> <li>• Detail is more than 1000 characters long</li> <li>• Time_start and time_end is not within 0000 and 2400</li> <li>• Time_end is earlier than time_start</li> </ul> <b>Access Error</b> when any of: <ul style="list-style-type: none"> <li>• Token is invalid</li> <li>• Token is not a participant of the event_id</li> </ul>
<b>team/calendar/remove</b>  Given an event_id for an event, remove from the channel/dm calendar.  This event will also be removed from the personal calendars of the members in channel/dm.	DELETE	<b>Parameters:</b> (token, event_id)  <b>Return Types:</b> {}	<b>Input Error</b> when any of: <ul style="list-style-type: none"> <li>• Event_id is invalid</li> <li>• Event_id is already removed</li> </ul> <b>Access Error</b> when any of: <ul style="list-style-type: none"> <li>• Token is invalid</li> <li>• Token is not a participant of the event_id</li> </ul>

## 2.2 | Conceptual Modelling (State)

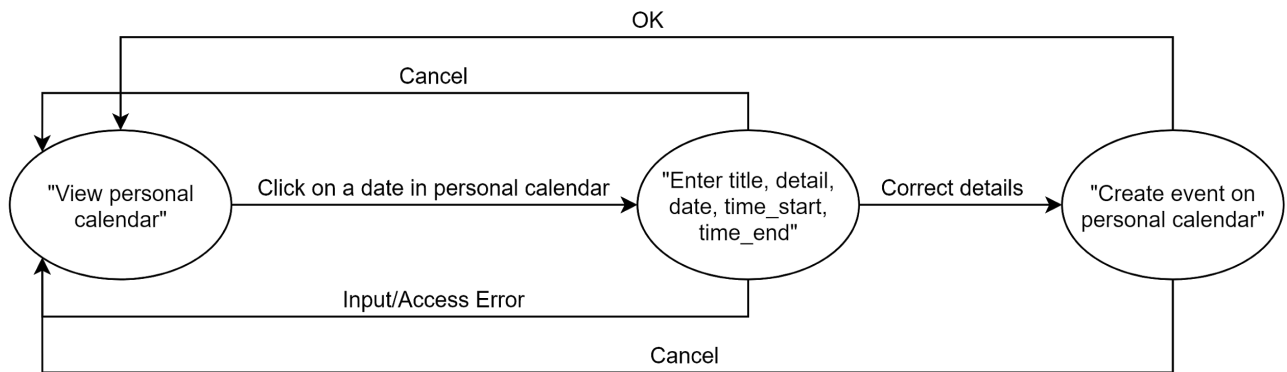


Figure 1a: State diagram for creating an event on a personal calendar

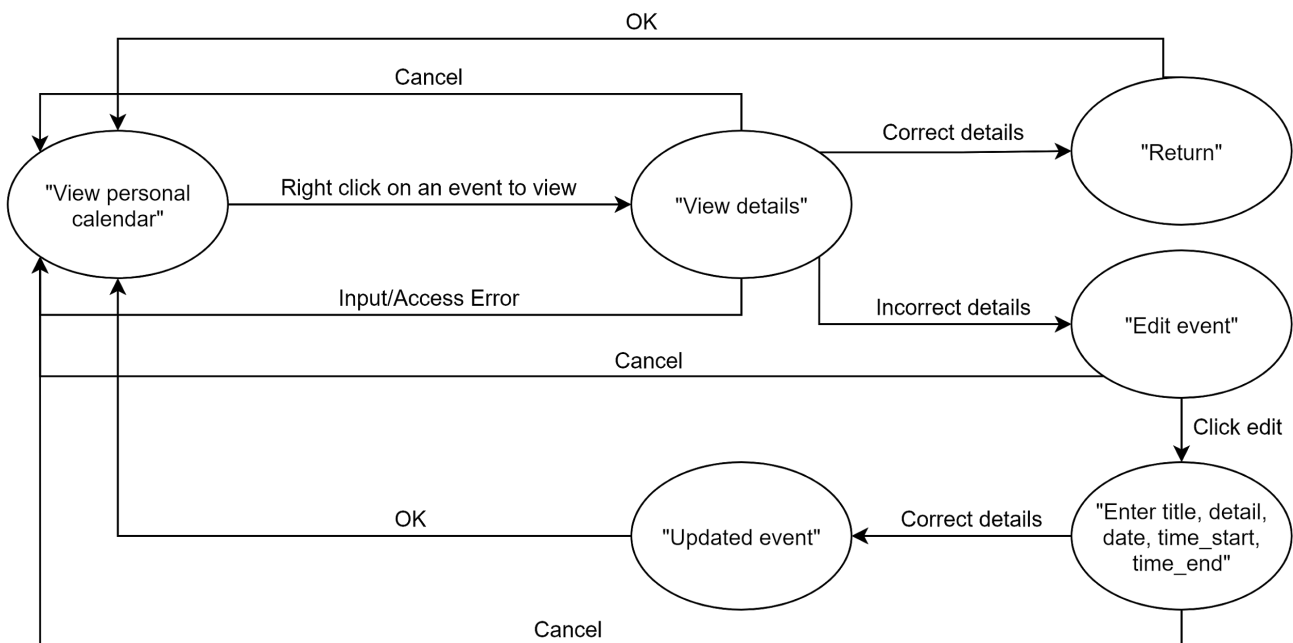


Figure 1b: State diagram for editing an event on a personal calendar

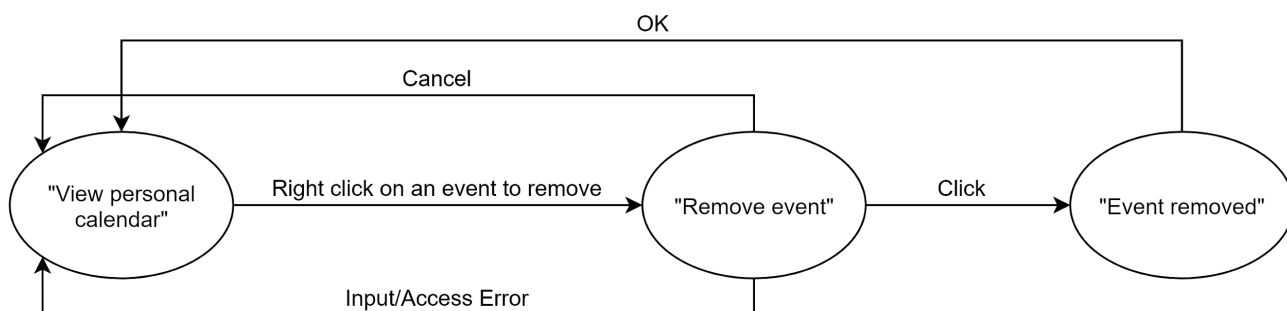


Figure 1c: State diagram for removing an event on a personal calendar

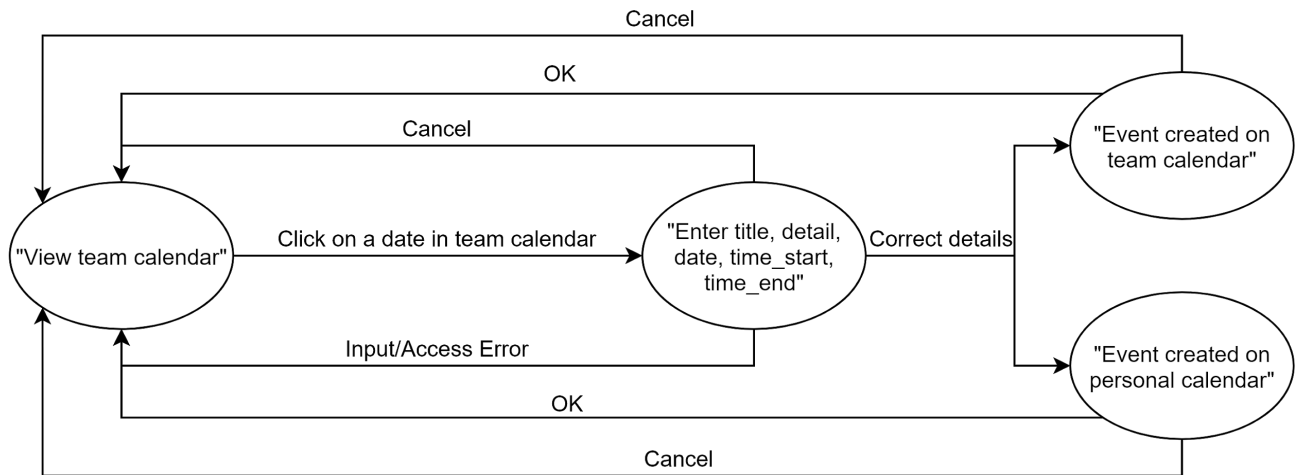


Figure 2a: State diagram for creating an event on a team calendar

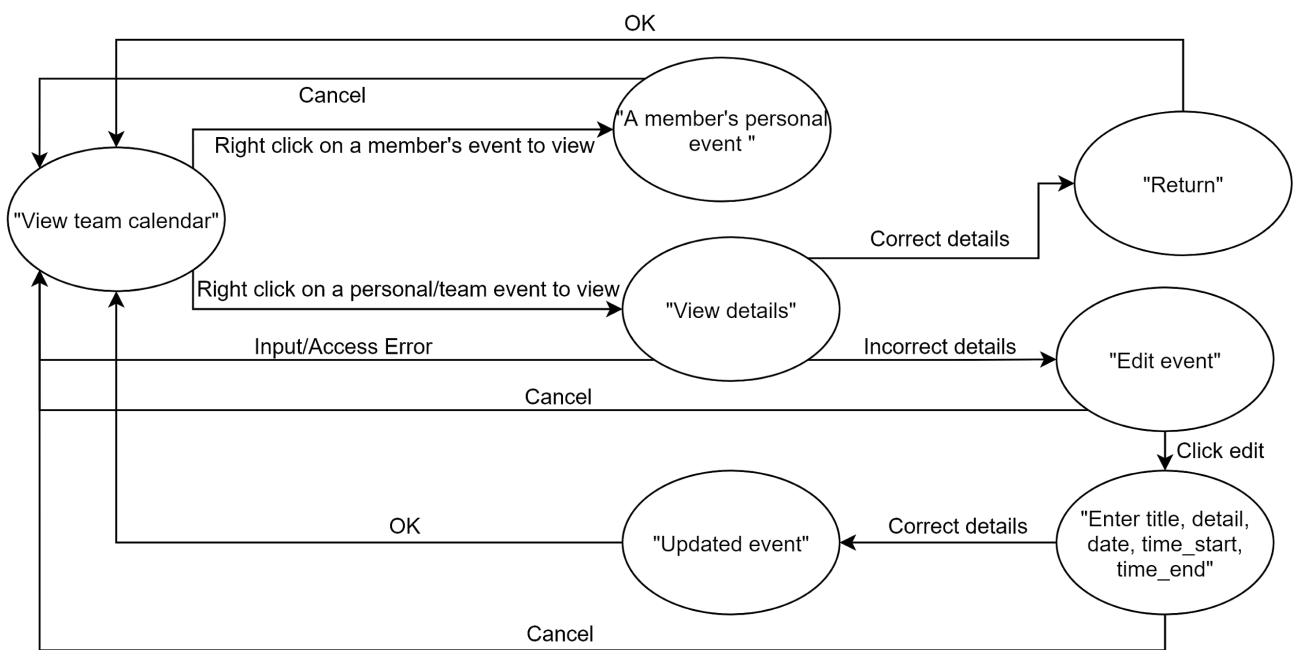


Figure 2b: State diagram for editing an event on a team calendar

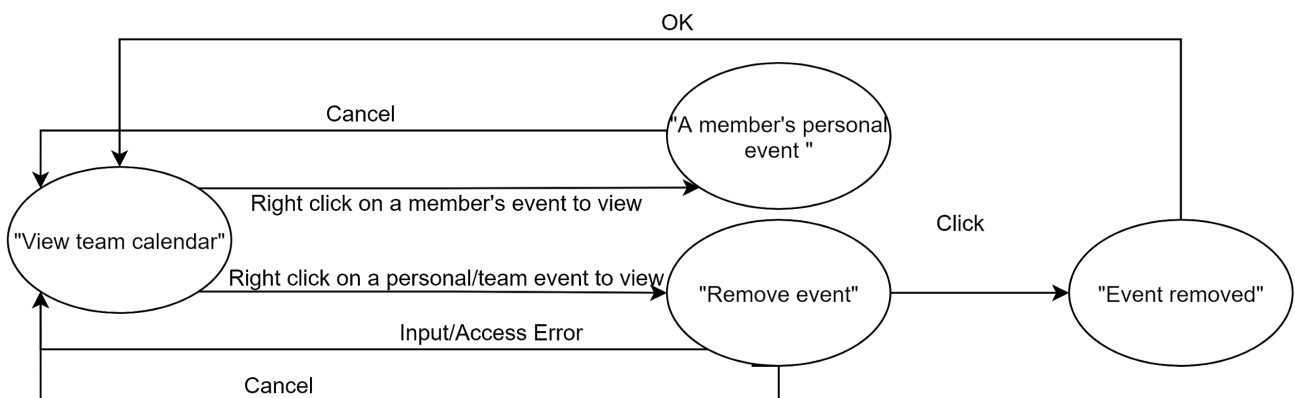


Figure 2c: State diagram for removing an event on a team calendar