Use Cases

for

Tutoria

Version 3.0

**Prepared by Nian Xiaodong, Tang Chi Ian,
Wang Junjie, Wang Michelle Yih-chyan**

Panacea Inc.

November 12, 2017

Revision History

Name	Date	Reason for Changes	Version
Nian Xiaodong, Tang Chi Ian, Wang Junjie, Wang Michelle Yih-chyan	October 11, 2017	Original	1.0
Nian Xiaodong, Tang Chi Ian, Wang Junjie, Wang Michelle Yih-chyan	October 28, 2017	Feedback from the client	2.0
Nian Xiaodong, Tang Chi Ian, Wang Junjie, Wang Michelle Yih-chyan	November 12, 2017	Refinements according to the implementation.	3.0

Use Case UC-1 Book a Session

Scope: Tutoria application
Level: User goal

Primary and Supporting Actors

Primary Actor: User

Description

Users want to book a tutorial session for their courses. They browse under the scope of the courses they are taking, enter a specific date, and find some available tutorial time slots for different tutors. Then they could choose their preferred time and tutor, and confirm the booking. Finally, the booked session will be added to the schedule of both students' calendar and that specific tutor's calendar.

Stakeholders and Interests

Student: Wants a clear view of the available time slots and tutors. Wants to ensure that the wallet system and the payments are carefully handled, especially in terms of the safety of the payments and the correct calculation of price. Wants a user-friendly way to submit a booking request. Wants to be clearly notified the time and tutor information of their booking.

Tutor: Wants to be clearly and quickly notified the time and student information of any booked session.

MyTutors: Wants an accurate record of booking history and transactions. Wants to ensure that the wallet system and the payments are carefully handled, especially in terms of the safety of the payments and the correct calculation of price.

Preconditions

PRE-1: User's identity has been authenticated.

Postconditions

POST-1: The booking selected by the user has been recorded.
POST-2: Email confirmations have been sent to the user, as well as to the tutor.
POST-3: The fee for the tutorial session (if any) is subtracted from the student and held temporarily in the system.
POST-4: A confirmation page has been shown to the user.

Main Success Scenario

1. The user enters the tutorial search page.
2. The system returns a list of available search criteria and the UI components for specifying these criteria, including tutor's name, the name of university, university course, subject tags, hourly rate range (minimum and maximum), whether tutor is contracted or private, and whether to show all tutors or only those with at least one available timeslot in the next 7 days.

3. The user specifies any number of the available search criteria and initializes a search.
4. The system returns a list of tutors and an overview of their profiles and timeslots that satisfied all the user's search criteria.
5. The user selects the desired tutor.
6. The system returns the detailed information about the tutor, and the available timeslots.
7. The user selects the desired time slot and submits a booking request.
8. The system locks the desired time slot, temporarily performs a status update on the time slot from available to locked, calculates the price, returns the price and others details of the booking, and asks the user to confirm the booking.
9. The user confirms the booking.
10. The system transfers the fee from the wallet to the internal system, performs a status update on the booking from locked to booked, records this booking and transaction, and then notifies the user that the booking was successful.
11. The system sends notifications to both student and tutor.

Extensions

***a.** At any time, the system fails:
To respond to the user request on the web page, display the web page successfully and return to the previous state before the mistake occurs.

1. The user can restart the website, log in, request recovery or reload to the prior state.
2. System reconstructs prior state

***b.** At any time, the administrator requests override:

1. The system enters Administrator-authorized mode.
2. The administrator performs one Administrator-mode operation (e.g. wallet balance change, resume on a booking process, void a booking...).
3. System returns to user-mode.

4a. No tutor satisfies all the search criteria specified by the user

1. The system notifies user "No available tutor", and asks the user to change the search criteria.
2. The criteria are restored at the search page.
3. The use case continues at step 2.

5b. The user wants to change the search criteria

1. The user changes the search criteria as desired and starts a new search.
2. The use case continues at step 2.

5b. The user wants to change the sort order of the results by hourly rate

1. The user clicks the menu bar of "price".
2. The system returns a sorted list of search results with the same search criteria.
3. The use case continues at step 5.

8a. Multiple users want to book one same session

1. The system will first lock the session for the first user as mentioned.
   1a. The first user can proceed to confirmation page accordingly.
   1b. The use case continues at step 9.
2. The system rejects other booking requests, and the other users will not be able to view that timeslot, and returns the users to previous step.

2a. The use case continues at step 6.

8b. If the user wants to book a session that is on the same day with any session he/she already has

1. The system notifies the user “this session cannot be booked due to the same day prohibition”, rejects the request and asks the user to return to prior steps.
2. The use case continues at step 6.

8c. The time slot user wants to book is within 24 hours before the session starts

1. The system determines this session cannot be booked, returns an error message to the user, and asks the user to return to prior step
2. The use case continues at step 6.

8d. The time slot user wants to book is blacked out by the tutor

1. The user will not be able to select that timeslot.
2. The use case continues at step 6.

9a. The user enters a coupon code for sessions that are not free

1. If the coupon code is valid.
   1a. The system recalculates the price, asks the user to confirm the details again.
   1b. The use case continues at step 9.
2. If the coupon code is invalid.
   2a. The notifies the users, asks the user to enter the coupon code or confirm the details again,
   2b. The use case continues at step 9.

9b. The user cancels the booking request.

1. The system performs a status update on the time slot from locked to available.
2. The use case continues at step 7.

9c. The user terminates the confirmation page.

1. The system will keep the session locked for this user, pending for further confirmation.
2. The user can go to “My Tutorials” to continue the unfinished booking. The use case continues at step 9.

10a. The wallet balance of user is less than the required payment

1. The system reports “not enough balance” to the user.
2. The system lets the user choose whether to cancel this booking or to add more balance.
3. If the user chooses to cancel the booking.
   3a. The system performs a status update from locked to available.
   3b. The use case continues at step 7.
4. If the user chooses to add more balance
   4a. then the system will pop up the payment page. After the payment, the user can try to confirm the booking again.
   4b. The use case continues at step 9.

11a. User or tutor cannot receive the confirmation email

1. They can still see the booked session from the calendar on their schedule page.
2. They can set up reminder emails to remind them every day of today’s schedule in case they miss any session.

Frequency of Occurrence

Almost continuous.

Business Rules

Business rules that apply to this use case are:
BR-1c, BR-1e, BR-1f, BR-1h;
BR-2a, BR-2b, BR-2c, BR-2d, BR-2f, BR-2g, BR-2h, BR-2i;
BR-3a, BR-3b;
BR-4a;
BR-5b, BR-5c;
BR-6a.

Special Requirements

1. Transactions from the user must be made and temporarily locked in a secured way.

Technology and Data Variations List

***a.** Administrator-mode is entered by logging in with the administrator account and typing in the password.
3a. A user input parser can be implemented to parse the course codes to match the format in our database.
5a. The user can change the date by typing or selecting from the pop-up calendar.
8a. Race condition must be avoided when multiple users want to book the same session.
9a. The system needs to identify valid and invalid coupon code.

Assumptions

None.

Open Issues

- What technique should be used to avoid race condition of booking a session?

Use Case UC-2 Cancel a Session

Scope: Tutoria application
Level: User goal

Primary and Supporting Actors

Primary Actor: User

Description

Student users want to cancel a tutorial session they have already booked. The users enter their schedule page and can see all the sessions they are going to attend in the future. Then users can click any session they have and choose to cancel it. The user then receives the refund if applicable.

Stakeholders and Interests

Student: Wants a clear review of all the sessions booked on the calendar. Wants a user-friendly way to submit a canceling request. Wants to get refunded after canceling the session if it charged money. Wants to get notified if they cancel a session successfully. Wants to ensure that the wallet system and the payments are carefully handled, especially in terms of the safety of the payments and the correct calculation of price.

Tutor: Wants to be clearly and quickly notified if a student has canceled a session taught by this tutor. Wants a clear review of his/her calendar after any cancellation.

MyTutors: Wants an accurate record of booking and canceling history. Wants to ensure that the wallet system and the payments are carefully handled, especially in terms of the safety of the payments and the correct calculation of price.

Preconditions

PRE-1: User’s identity has been authenticated.

Postconditions

POST-1: The canceled session by the user has been recorded, and marked as available to all student users.
POST-2: Email confirmations have been sent to the user, as well as the to the tutor.
POST-3: The fee for the tutorial session (if any) is returned to the user.
POST-4: A confirmation page has been shown to the user.

Main Success Scenario

1. The user enters schedule page to view all his/her booked sessions.
2. The system displays all the booked sessions of that user in a calendar.
3. The user chooses on one of his/her booked session on the calendar.
4. The system returns the detailed information of the booked session.
5. The user chooses to submit a cancelation request.
6. The system calculates the amount of refund, returns the amount together with other details about the session to the user, and asks the user to confirm the cancelation request.
7. The user confirms the cancelation request.

8. The system refunds to the amount to the user, records this cancellation and transaction and then notifies that the cancelation was successful.
9. The system sends notifications to both student and tutor.
10. The system marks the canceled session available to be booked by all users.

Extensions

***a.** At any time, the system fails:
To respond to a user request on the web page, display the web page successfully and return to the previous state before the mistake occurs.

1. The user can restart the website, log in, request recovery or reload to the prior state.
2. The system reconstructs prior state

***b.** At any time, the administrator requests override:

1. The system enters Administrator-authorized mode.
2. The administrator performs one Administrator-mode operation (e.g. wallet balance change, resume on a canceling process, void a cancellation,...).
3. The system returns to user-mode.

5a. The session user chooses to cancel is going to begin within 24 hours

1. The system notices the state of this session as “committed”, and rejects the cancelation request.
2. The system returns an error message to the user, mentioning that the session cannot be canceled, and asks the user to return to prior steps.
3. The use case continues at step 3.

7a. The user terminates the cancelation confirmation.

1. The system returns the user to the previous page.
2. The use case continues at step 3.

9a. User or tutor cannot receive the confirmation email

1. They can still see the canceled session disappeared on their calendar.
2. They can set up reminder emails that remind them every day of today’s schedule in case they miss any notification.

Frequency of Occurrence

Almost continuous.

Business Rules

Business rules that apply to this use case are:
BR-2e, BR-2j.

Special Requirements

1. If a session is not able to be cancelled at the time of viewing, the “cancel” button on the information page is displayed by a color different from other options, and it is not clickable.
2. The amount of refund money must be same as the amount charged when the session is booked.

3. Users must remain Internet connection during the time they are booking a session.

Technology and Data Variations List

***a.** Administrator-mode is entered by logging in with an administrator account and typing in the password.
5a. The system has to calculate the time remaining to the beginning of a session in order to determine this session can be canceled or not.

Assumptions

1. Assume users know how much time remaining to the beginning of the session they want to cancel.

Open Issues

- How to update the “cancel” button according to the current time and the beginning time of a specific session?

Use Case UC-3 Begin All Sessions

Scope: Tutoria application
Level: User goal

Primary and Supporting Actors

Primary Actor: Administrators from MyTutors
Supporting Actors: None

Trigger

Every half an hour, this use case is triggered.

Description

All booked sessions are confirmed and cannot be modified, and available sessions are closed under normal circumstances 24 hours before the starting time of the corresponding tutorial session. This involves the finalization of tutorial sessions, the change in state from ‘booked’ to ‘committed’, as well as the change in state from ‘available’ to ‘unavailable’ for sessions that are not booked.

Stakeholders and Interests

Student: Wants a final confirmation of the booking, where there will not be any changes under normal circumstances.

Tutor: Wants to know the time beyond which the students cannot change their bookings, and hence a final confirmation on holding the session.

Preconditions

PRE-1: There exist booked sessions in the system.

Postconditions

POST-1: All the booked sessions which are going to be held within 24 hours later are changed to the ‘committed’ state which cannot be further modified.
POST-2: All available sessions which are to be held within 24 hours later are changed to ‘unavailable’ state which cannot be booked.
POST-3: Reminder email confirmations have been sent to the user, as well as the to the tutor.

Main Success Scenario

1. All the sessions which are to be start in 24 hours are identified, and then are locked where the users are unable to make further changes
2. The system performs a status update from ‘booked’ to ‘committed’ on the booked sessions, and a status update from ‘available’ to ‘unavailable’ on the available sessions.
3. The system sends notifications to both student and tutor for the changes.

Extensions

***a.** At any time, the system fails:

1. The system reverts to the previous state before the mistake occurs.

***b.** At any time, Administrator requests override:

1. The system enters the administrator-authorized mode.
2. Administrator performs one Administrator-mode operation (e.g. Manually changing the state of sessions, cancellation of sessions...).
3. System returns to user-mode.

1a. There does not exist any booked sessions which are to be held within 24 hours

1. The system finishes the state updating phase immediately and returns to normal operation

3a. User or tutor cannot receive the confirmation email

1. They can still see the booked session from the calendar on their schedule page.
2. They can set up reminder emails to remind them every day of today's schedule in case they miss any session.

Frequency of Occurrence

Every half an hour.

Business Rules

Business rules that apply to this use case are:
BR-2b, BR-2c, BR-2e.

Special Requirements

1. The state of the tutorial session is visible to users in the UI design.

Technology and Data Variations List

***a.** The administrator-mode is entered by logging in with Administrator account and typing in the password.

1a. Race condition must be avoided when the system tries to lock the sessions and users want to book or cancel the sessions.

Assumptions

1. The system is running normally.

Open Issues

- How to avoid inconsistency in the system states if the administrator overrides?

Use Case UC-4 End All Sessions

Scope: Tutoria application
Level: User goal

Primary and Supporting Actors

Primary Actor: Administrators from MyTutors
Supporting Actors: None

Trigger

Every half an hour, this use case is triggered.

Description

At the time when all sessions of that time slot finished, these committed sessions will be treated as ended and completed. This involves the payment transaction of tutorial sessions, reviewing function activation and an invitation to students who have taken these sessions, as well as the change in state from ‘committed’ to ‘ended’.

Stakeholders and Interests

Students: Wants a clear view of their finished tutorial sessions and payment history. Wants to review the session, offering information for students to compare different tutors.

Tutors: Wants a clear view of their finished tutorial sessions history. For private tutors, they want to ensure that the wallet system and the payments are carefully handled, especially in terms of the safety of the payments and the correct calculation of price.

MyTutors: Wants an accurate record of booking history and transactions. Wants to ensure that the wallet system and the payments are carefully handled, especially in terms of the safety of the payments and the correct calculation of price.

Preconditions

None.

Postconditions

POST-1: All the ‘committed’ sessions that are ended are changed to the ‘ended’ state.
POST-2: Reviewing sections for the finished tutorials are activated, and reviewing invitation emails are sent to students.
POST-3: Tutorial payments are transferred to tutor.

Main Success Scenario

1. All the committed sessions which are completed at the time of trigger are identified.
2. The system a status update from ‘committed’ to ‘ended’ on the sessions.

3. The system calculates the corresponding commission for each tutorial session, and transfers all the commissions to MyTutors.
4. The system records this transaction and the completion of the tutorials, and sends notifications to the corresponding tutors.
5. The system activates the review submission section for the tutorials for the students, and sends notifications to the corresponding students.

Extensions

***a.** At any time, the system fails:

1. The system reverts to the previous state before the mistake occurs.

***b.** At any time, the aministrator requests override:

1. The system enters the administrator-authorized mode.
2. The administrator performs one administrator-mode operation (e.g. Manually changing the state of sessions, cancellation of sessions,...).
3. The system returns to user-mode.

1a. There does not exist any committed sessions which ends at the time when the case triggered:

1. The system finishes the state updating phase immediately and returns to normal operation

4a. The student cannot receive the review invitation email:

1. They can still see the history tutorial in the dashboard and submit a review.

4a. Tutors do not receive the payment notification:

1. They can check the transaction history on their wallet page and see the latest payment
   1a. If they see this payment, they do not need to do anything.
   1b. If they do not see this payment, they can contact the administrator and request a payment.

Frequency of Occurrence

Almost continuous.

Business Rules

Business rules that apply to this use case are:
BR-1a, BR-1b, BR-1c, BR-1d.

Special Requirements

1. The passwords of the users must be stored in a secure way in the system.

Technology and Data Variations List

***a.** The administrator-mode is entered by logging in with an administrator account and typing in the password.
3a. Transactions to the user must be made in a secure way.

Assumptions

1. The system is running normally

Open Issues

- How to deal with the payment transaction failure when the session ends?

Use Case UC-5 Sign Up

Scope: Tutoria application
Level: User goal

Primary and Supporting Actors

Primary Actor: User
Supporting Actors: None

Description

At the time when a user wants to obtain access to the Tutoria system, the users would be asked to register themselves on the system. This involves the setup of their username and password, and the subsequent authentication.

Stakeholders and Interests

Students: Wants to register as a student. Wants to be able to book sessions on the system. Wants to review the information provided on the system.

Tutors: Wants to register as a contracted or private tutor. Wants to create sessions on the system. Wants to offer services on the system.

MyTutors: Wants to control access to Tutoria through user authentication based on username and password. Wants to notify the user a token when users forget their passwords.

Preconditions

PRE-1: Users have valid Email addresses.

Postconditions

POST-1: All the usernames, passwords and other information are recorded by the system.

Main Success Scenario

1. The user enters the log-in page and chooses to sign-up for Tutoria.
2. The system returns the list of information to be filled in, including username, password, confirmation password, and other user information, and the corresponding UI components.
3. The user inputs all the information and submits.
4. The system notifies the user that the registration is successful, and sends an email confirmation message to the user.

Extensions

***a.** At any time, the system fails:
   1. The system reverts to the previous state before the mistake occurs.

***b.** At any time, the administrator requests override:

1. The system enters the administrator-authorized mode.
2. The administrator performs one administrator-mode operation (e.g. Manually changing the information of users, remove accounts, ...).
3. The system returns to user-mode.

3a. The user enters a username and email which match one of the stored profiles in the system

1. The system returns an error message which states that there is a matching profile and asks whether the user wants to re-enter new credentials or to try to log in
2. Previously entered data except passwords are restored.
3. The user case continues at step 2.

3b. The user enters a username which has been used already, but the email does not match the username in the system.

1. The system returns an error message, and asks the user to choose another username.
2. Previously entered data except passwords are restored.
3. The user case continues at step 2.

3c. The user does not provide all the necessary information.

1. The system returns an error message, and asks the user to provide the missing information.
2. Previously entered data except passwords are restored.
3. The user case continues at step 2.

3c. The password provided by the user does not pass the security check.

1. The system returns an error message, and asks the user to provide another password following the security rules.
2. Previously entered data except passwords are restored.
3. The user case continues at step 2.

3d. The user enters the password and the confirmation password differently.

1. The system returns an error message, and asks the user to re-enter the password and confirmation password.
2. Previously entered data except passwords are restored.
3. The user case continues at step 2.

3e. Any of the information provided by the user is malformed or invalid.

1. The system returns an error message, and asks the user to re-enter the corresponding information.
2. Previously entered data except passwords are restored.
3. The user case continues at step 2.

Frequency of Occurrence

Almost continuously.

Business Rules

Business rules that apply to this use case are:
BR-2b, BR-2c, BR-2j,
BR-7a.

Special Requirements

None.

Technology and Data Variations List

3a. The system can verify passwords the user created if they match the NIST criteria.

Assumptions

1. The system is running normally

Open Issues

- How to verify the authenticity of emails provided by users?

Use Case UC-6 Forget Password

Scope: Tutoria application
Level: User goal

Primary and Supporting Actors

Primary Actor: User
Supporting Actors: None

Description

At the time when a user wants to log in to the Tutoria system, the users would be asked to provide their usernames and passwords on the system, or to provide their user identifications to verify their identities. This involves the checking of the combination of usernames and passwords, changing of their passwords, password-forgetting email sending, and the subsequent authentication.

Stakeholders and Interests

Students: Wants to regain access to the system. Wants to be able to book sessions on the system. Wants to review the information provided on the system.

Tutors: Wants regain access and log in as a contracted or private tutor. Wants to create sessions on the system. Wants to offer services on the system.

MyTutors: Wants to regain the control access to Tutoria through user authentication based on username and password. Wants to notify the user a token when users forget their passwords.

Preconditions

None.

Postconditions

POST-1: All the password changes and other information are updated and recorded by the system.

Main Success Scenario

1. The user enters the log-in page and chooses to initialize a lost password request.
2. The system returns the list of information required for such request, including username and email.
3. The user enters the required information and submits the request.
4. The system verifies the information, sends a reset token to the user’s email, and asks the user to provide the password reset token.
5. The user submits the reset token.
6. The system verifies the token, and asks the user to provide a new password.
7. The user submits the new password.
8. The system records the new password, notifies the user that the rest is successful, and sends an email confirmation to the user.

Extensions

***a.** At any time, the system fails:

1. The system reverts to the previous state before the mistake occurs.

***b.** At any time, the administrator requests override:

1. The system enters the administrator-authorized mode.
2. The administrator performs one administrator-mode operation (e.g. Manually changing the information of users, remove accounts, ...).
3. The system returns to user-mode.

4a. The information provided by the user does not match any record in the system.

1. The system returns an error message, and asks the user to re-enter the information.
2. The user case continues at step 2.

5a. The user does not receive the token from email.

1. The user goes back to password-reset requesting page and requests the token again from the system.

6a. The user does not provide a valid reset token.

1. The system returns an error message, and asks the user to re-enter the token.
2. The user case continues at step 5.

8a. The password provided by the user does not pass the security check.

1. The system returns an error message, and asks the user to provide another password following the security rules.
2. The user case continues at step 7.

Frequency of Occurrence

Almost continuously.

Business Rules

Business rules that apply to this use case are:
BR-1a, BR-1b, BR-1c,

Special Requirements

None.

Technology and Data Variations List

4a. The system can generate token and email to the user’s email.

Assumptions

1. The system is running normally

Open Issues

- How to prevent the passwords and account information being leaked?

Use Case UC-7 Modify Profile

A logged-in user would like to modify the profile. The user clicks the “profile” button. The “profile” button offers a selection menu with a “view profile” button. The user views its profile by clicking the button. The user clicks the “modify” button to modify the profile of the user. The user completes the modification to its profile and clicks “submit” button. The system receives the new profile information, and updates the database as well as the corresponding profile shown on the interface. The user returns back to the “view profile” interface.

Use Case UC-8 Add Balance & Withdraw Balance

A logged-in user would like to add balance to the wallet. The user pushes the “profile” button. The “profile” button offers a selection menu with “wallet” button. The user clicks the “wallet” button. The website which shows the present balance is displayed. The user clicks the “add balance” button to add or “withdraw balance” button to withdraw the desired amount. The system updates the balance of the user. The website shows the changed balance of the user.

Use Case UC-9 Blackout Unavailable Time Slot

A logged-in user as a tutor would like to blackout unavailable time slot. The user clicks the “schedule” button. The schedule page shows the current weekly schedule of the user. The user can switch to the previous week and the following week by clicking the “previous week” and “next week” buttons. The user blacks out timeslots for which they are not available during the current and the following week. The system receives the updated schedule and updates the database as well as the web pages showing the schedule. The user returns back to the schedule page.

Use Case UC-10 Tutor Review

A logged-in user as a student completes a tutorial session. The student clicks the “my course” button. The page shows all the booked, confirmed and completed tutorial sessions. The student chooses the completed tutorial sessions and gives a course review to the completed tutorial clicking the completed tutorial. The page shows the completed tutorial information and offers a text field. The student writes the course review in the text field and submits it. The system records the course review and updates the received course review to the database. The course status would be changed from ended to reviewed.

Use Case List

Primary Actor	***Use Case***	***Priority****
Student users	UC-1 Book a Session	1
Student users	UC-2 Cancel a Session	2
Online booking system	UC-3 Begin All Sessions	3
Online booking system	UC-4 End All Sessions	4
All users	UC-5 Sign Up	6
All users	UC-6 Forget Password	7
All users	UC-7 Modify Profile	9
All users	UC-8 Add Balance & Withdraw Balance	8
Tutor users	UC-9 Blackout Unavailable Time Slot	5
Student users	UC-10 Tutor Review	10

**Priority: the smaller the number, the higher the priority.*