

Use case List

Login Process

- L1 - User login
- L2 - User logout
- L3 - Reset password
- L4 - User signup

User Management

- U1 - Update user role (superstudent)
- U2 – Update user role (admin)
- U3 - Update user profile (admin/superstudent)
- U4 – Update your profile
- U5 – Whitelist user email
- U6 – Fire someone (superstudent)
- U7 – Fire someone (admin)

Schedule

- S1 - View schedule (Student)
- S2 - View schedule (Superuser/admin)
- S3 - Add to schedule
- S4 – Change to schedule
- S5 – Remove from schedule

Garbage collection schedule

- G1 - View garbage collection schedule of a building (Admin/superstudent)
- G2 - View garbage collection schedule of a building (Syndic)
- G3 - Add garbage collection info to building
- G4 - Remove garbage collection info from building
- G5 - Change garbage collection schedule of building
- G6 - Duplicate garbage schedule for building

Tour

- T1 – View a tour
- T2 – Create a new tour (superstudent/admin)
- T3 – Update a tour
- T4 – Remove a tour

Mail

- M1 – View a mail template (Admin/superstudent)
- M2 – Create a new mail template (Admin/ superstudent)
- M3 – Change an existing mail template (Admin/superstudent)
- M4 – Send a mail (Admin/superstudent)

- M5 – Remove mail template (Admin/superstudent)

Analysis

- A1 – View analysis of a student
- A2 – View analysis of a building
- A3 – View analysis of a tour
- A4 – View analysis of total working hours

Dashboard

- D1 – View all ongoing tours (Admin/superstudent)
- D2 - View all tours, filtered on certain information (Admin/superstudent)
- D3 – View specific information about a tour (Admin/superstudent)
- D4 – View specific information about a building on a tour (Admin/superstudent)
- D5 – View tours planned this week (Student)
- D6 - View Information about previous tours (Student)
- D7 – Start a tour (Student)
- D8 – Do a building on a tour (Student)

Building

- B1 – View a building (superstudent/admin)
- B2 – View a building (Syndicus)
- B3 – Add a building (Superstudent/admin)
- B4 – Remove a building (superstudent/admin)
- B5 – Change a building (superstudent/admin)
- B6 – Upload a manual for a building (Superstudent/admin/Syndicus)

Use Cases

Login Process

UC-Name	L1 - User Login
Short informal description:	Logging in to the system
Primary Actor:	Student/Superstudent/Admin/Syndicus (Users)
Pre-conditions:	A) User has an existing account B) User is not currently logged in
Normal Flow:	1. User enters its credentials (email + password) 2. User clicks on <i>Login</i> button 3. System validates credentials 4. User is logged in and redirected to the main page
Postconditions:	A) User is currently logged in B) User can further navigate the web application, depending on its role
Alternative Flows:	System rejects credentials 1. System returns an error message 2. Jumps to step 1 of <i>Normal Flow</i>

UC-Name	L2 – User Logout
Short informal description:	Logging out of the system
Primary Actor:	Student/Superstudent/Admin/Syndicus (Users)
Pre-conditions:	A) User is currently logged in
Normal Flow:	1. User clicks on <i>Profile</i> button 2. User clicks on <i>Log out</i> button 3. Pop up message asks for confirmation 4. User confirms 5. The user gets logged out by the system 6. The user is being redirected to the login screen
Postconditions:	A) User is not currently logged in B) User cannot currently navigate the web application
Alternative Flows:	User denies the confirmation message in step 3 of <i>Normal Flow</i> 1. Confirmation message disappears

UC-Name	L3 – Reset Password
Short informal description:	Resetting your password
Primary Actor:	Student/Superstudent/Admin/Syndicus (Users)
Pre-conditions:	A) User has an existing account B) User is not currently logged in C) User is on the login screen D) User has an old (unknown) password X
Normal Flow:	1. Click the <i>Reset password</i> button

	<ol style="list-style-type: none"> 2. User is redirected to a password reset page 3. Fill in the email of your existing account 4. Click the <i>Confirm</i> button 5. Success message appears, stating that an email has been sent to the user 6. User receives a link through email to reset their password 7. The <i>Confirm</i> button is disabled 8. Fill in your new password 9. Fill in your new password again as confirmation 10. The <i>Confirm</i> button is enabled 11. Click the <i>Confirm</i> button 12. System uses token of the link to update password in database 13. Success message appears 14. User is redirected to the login page
Postconditions:	<ol style="list-style-type: none"> A) User is still not currently logged in B) User has a new password Y
Alternative Flows:	<p>User enters different passwords in both password fields</p> <ol style="list-style-type: none"> 1. Visual feedback appears on the confirmation field (e.g., red border or a text) 2. The <i>Confirm</i> button stays disabled 3. Jumps to step 9 of <i>Normal Flow</i>

UC-Name	L4 – Sign Up
Short informal description:	Registering a new account in our application
Primary Actor:	Student/Superstudent/Admin/Syndicus (Users)
Pre-conditions:	<ol style="list-style-type: none"> A) Email is not yet registered in our application B) User is on the login screen
Normal Flow:	<ol style="list-style-type: none"> 1. Click <i>Sign Up</i> button 2. User is redirected to a Sign-Up page 3. Fill in all input fields with correct information 4. When all required fields are filled in successfully, a <i>Create Account</i> button is enabled 5. User clicks the <i>Create Account</i> button 6. System stores all the information in the database 7. Success message appears 8. User is redirected to welcome page, with empty schedule and no tours so far 9. Admin is notified that this account is created
Postconditions:	<ol style="list-style-type: none"> A) Email and user are now registered in our application A) User is on the welcome page
Alternative Flows:	<p>User is first ever created user</p> <ol style="list-style-type: none"> 1. Same steps as <i>Normal Flow</i> 2. User is automatically granted the <i>admin</i> role <p>Input fields in step 3 in <i>Normal Flow</i> are not filled in correctly</p> <ol style="list-style-type: none"> 1. The <i>Create Account</i> button remains disabled 2. Visual feedback appears on the incorrect fields 3. Jumps to step 3 in <i>Normal Flow</i>

User Management

UC-Name	U1 - Update user role (Superstudent)
Short informal description:	Update the role assigned to a specific user as a superstudent. A superstudent may promote students to superstudents, but can't degrade
Primary Actor:	Superstudent
Pre-conditions:	A) User is on welcome page B) There are other existing students
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Students</i> tab 2. User is redirected to the <i>users</i> screen 3. User views a display off all students that he/she is allowed to monitor 4. Click on the student you want to change the role of 5. The user is redirected to the page of the specified student 6. Click on the <i>Edit Role</i> button 7. The role becomes a multiselect button 8. Select the role you want to promote to (superstudent) 9. Pop up appears: <i>Are you sure? This action can only be undone by an admin</i> 10. Click on <i>Confirm</i> 11. The database is updated with the new role
Postconditions:	A) The specified student is now a superstudent
Alternative Flows:	You want to degrade another superstudent to a student <ol style="list-style-type: none"> 1. Ask an admin to follow flow U2

UC-Name	U2 - Update user role (Admin)
Short informal description:	Update the role assigned to a specific user as an admin. An admin may promote and degrade any user as he/she sees fit. To degrade/promote an admin, see alternative flows
Primary Actor:	Admin
Pre-conditions:	A) User is on welcome page B) There are other existing students/superstudents
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Students</i> tab 2. User is redirected to the <i>users</i> screen 3. User views a display off all students that he/she is allowed to monitor 4. Click on the student you want to change the role of 5. The user is redirected to the page of the specified student 6. Click on the <i>Edit Role</i> button 7. The role becomes a multiselect button 8. Select the role you want to change to 9. Pop up appears: <i>Are you sure? This action can only be undone by an admin</i> 10. Click on <i>Confirm</i> 11. The database is updated with the new role
Postconditions:	B) The role of the specified student is now updated

Alternative Flows:	You want to promote a user to an admin <ol style="list-style-type: none"> 1. Normal flow 1-8 2. Select the admin role 3. All admins are notified of this and must approve this operation 4. Jumps to normal flow 9 5. The database is updated if and only if all admins approved
--------------------	---

UC-Name	U3 - Update user profile (Admin/Superstudent)
Short informal description:	Update any profile information of a specified student
Primary Actor:	Admin/superstudent
Pre-conditions:	C) User is on welcome page D) There are other existing students
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Students</i> tab 2. User is redirected to the <i>users</i> screen 3. User views a display off all students that he/she is allowed to monitor 4. Click on the student which profile information you want to change 5. The user is redirected to the page of the specified student 6. Click on the <i>Edit</i> button 7. All fields become input fields 8. Update all the information you intended to 9. Click on <i>Save changes</i> 10. The database is updated with the new information about the user 11. The user is redirected to the profile page of the specified student, with updated information
Postconditions:	C) The information of the specified student is now updated
Alternative Flows:	You want to update information of a superstudent/admin <ol style="list-style-type: none"> 1. Kindly ask said person to do it themselves

UC-Name	U4 - Update your profile
Short informal description:	Update your own profile information
Primary Actor:	Student/Superstudent/Admin/Syndicus (Users)
Pre-conditions:	A) User is on welcome page B) User has an existing account and is logged in
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Profile</i> tab 2. User is redirected to his profile screen 3. Click on the <i>Edit</i> button 4. All fields that the user is allowed to change become input fields. This is dependent on his/her role 5. Update all the information you intended to 6. Click on <i>Save changes</i> 7. The database is updated with the new information about the user

	8. The user is redirected to its profile page, with updated information
Postconditions:	A) The profile information of the user is updated accordingly
Alternative Flows:	As a student/syndicus you want to update information that you're not allowed to (e.g., email) 1. Inform a superstudent/admin to change this for you

UC-Name	U5 – Whitelist user email
Short informal description:	Allow a certain email to be able to create an account
Primary Actor:	Admin/superstudent
Pre-conditions:	A) User is on welcome page B) The email is not yet whitelisted C) The email is not allowed to make an account
Normal Flow:	1. Click on <i>Students</i> tab 2. User is redirected to the <i>users</i> screen 3. User views a display of all students that he/she is allowed to monitor 4. Click on the <i>Whitelist</i> button 5. The user is redirected to the whitelist screen 6. Click the <i>add</i> button 7. An input field appears 8. Fill in the email you want to whitelist 9. Click <i>Confirm</i> 10. A mail is sent to the specified email, including a verification key 11. The combination email, verification key is stored in the database 12. A popup appears stating that the operation was successful 13. The user is redirected to the whitelist screen, now with the new email added
Postconditions:	A) The email is now able to make an account, when providing the verification key
Alternative Flows:	/

UC-Name	U6 – Fire someone (superstudent)
Short informal description:	Fire a student
Primary Actor:	Superstudent
Pre-conditions:	A) User is on welcome page B) There are existing students C) The student isn't yet fired
Normal Flow:	1. Click on <i>Students</i> tab 2. User is redirected to the <i>users</i> screen 3. User views a display of all students that he/she is allowed to monitor 4. Click on the student you want to fire 5. The user is redirected to the page of the specified student 6. Click the <i>Fire</i> button

	<ol style="list-style-type: none"> 7. A popup appears: <i>Are you sure you want to fire this user?</i> 8. Click on <i>Confirm</i> 9. The student is set as inactive in the database 10. The user is redirected to the <i>users</i> screen. All fired students are displayed as semi-transparent
Postconditions:	B) The student is set as inactive (can't access his/her account anymore)
Alternative Flows:	/

UC-Name	U7 – Fire someone (admin)
Short informal description:	Fire any user
Primary Actor:	Admin
Pre-conditions:	<ol style="list-style-type: none"> A) User is on welcome page B) There are existing students C) The student isn't yet fired
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Students</i> tab 2. User is redirected to the <i>users</i> screen 3. User views a display off all students that he/she is allowed to monitor 4. Click on the person you want to fire 5. The user is redirected to the page of the specified person 6. Click the <i>Fire</i> button 7. A popup appears: <i>Are you sure you want to fire this user?</i> 8. Click on <i>Confirm</i> 9. The student is set as inactive in the database 10. The user is redirected to the <i>users</i> screen. All fired (super)students are displayed as semi-transparent
Postconditions:	C) The student is set as inactive (can't access his/her account anymore)
Alternative Flows:	<p>You want to fire an admin</p> <ol style="list-style-type: none"> 1. Normal flow 1-8 2. All admins (except for the one being fired) are notified and must approve this operation 3. Jumps to Normal flow 9 if and only if all admins have approved

Schedule

UC-Name	S1 – View schedule (Student)
Short informal description:	View the schedule of a specific student
Primary Actor:	Student
Pre-conditions:	A) Student is on the welcome page B) Student has a verified account and is logged in
Normal Flow:	1. Click <i>schedule</i> button 2. The student is redirected to the schedule page 3. The student gets a list of all the tours he is assigned in the future with a date 4. The student gets a list of all the tours he was assigned in the past.
Postconditions:	A) Student can further navigate the application
Alternative Flows:	Student doesn't have assigned tours: <ol style="list-style-type: none"> Normal flow 1-3 The list of all the assigned tours is empty Student has not done tours: <ol style="list-style-type: none"> Normal flow 1-3 The list of all the tours in the past will be empty

UC-Name	S2 – View schedule (Superstudent/admin)
Short informal description:	View the schedule of tours for user with role admin or superstudent
Primary Actor:	Superstudent/admin
Pre-conditions:	A) Superstudent/admin is on the welcome page B) Superstudent/admin has a verified account and is logged in
Normal Flow:	1. Click <i>schedule</i> button 2. The superstudent/admin is redirected to the schedule page 3. The user gets a list of all the tours with date that are assigned to students or other users. 4. The user gets a list of all the tours that were assigned in the past.
Postconditions:	A) User can further navigate the application
Alternative Flows:	If there aren't any assigned tours: <ol style="list-style-type: none"> Normal flow 1-3 The list of all the assigned tours is empty There are no tours in the archive: <ol style="list-style-type: none"> Normal flow 1-3 The list of all the tours in the past will be empty

UC-Name	S3 – Add to schedule
Short informal description:	Add and assign a tour to the schedule as admin or superstudent
Primary Actor:	Superstudent/admin
Pre-conditions:	A) The user must be logged in with a verified account

	B) The user must have the role admin or superstudent C) The user is on the welcome page of the site.
Normal Flow:	1. Click <i>schedule</i> button 2. The user is redirected to the <i>schedule</i> page 3. The user presses the <i>add to schedule button</i> 4. Select the tour that needs to be added to the schedule 5. Select the student that needs to do the tour 6. Select the date when the student needs to do the tour 7. Click the <i>add to Schedule button</i> 8. The user is redirected to the main schedule page
Postconditions:	A) User views the schedule with the added tour/student/date combination B) The user can further navigate the application
Alternative Flows:	The student is already assigned to that tour on that date: 1. Follow normal flow 1-7 2. An error displays which states that this entry already exists The tour doesn't exist yet: 1. Create a new tour with buildings (see T2)

UC-Name	S4 – Change to schedule
Short informal description:	Change a tour assignment in the schedule
Primary Actor:	Superstudent/admin
Pre-conditions:	A) The user must be logged in with a verified account B) The user must have the role admin or superstudent C) The user is on the welcome page of the application
Normal Flow:	1. Click the <i>schedule</i> button 2. The user is redirected to the <i>schedule</i> page 3. The user gets a view of the schedule (calendar) and can click on the assigned tours with student and date in the calendar 4. The user clicks on the tour and student combination that needs to be changed 5. The user is redirected to the <i>update schedule</i> page 6. The user changes the information of the select tour, student, date combination. 7. The user clicks the <i>update</i> button 8. The user is redirected back to the updated schedule
Postconditions:	A) The user views the updated schedule B) The user can further navigate the application
Alternative Flows:	The tour that needs to be changed has not yet been added to the schedule 1. Follow use case (S3) The tour that needs to be deleted is currently active: 1. Follow step 1-5 2. The user gets a message the tour is currently active, the user can't change any information nor delete

UC-Name	S5 – Remove from schedule
---------	---------------------------

Short informal description:	Remove a tour assignment in the schedule
Primary Actor:	Superstudent/ admin
Pre-conditions:	A) The user must be logged in with a verified account B) The user must have the role admin or superstudent C) The user is on the welcome page of the application
Normal Flow:	1. Click the <i>schedule</i> button 2. The user is redirected to the <i>schedule</i> page 3. The user gets a view of the schedule (calendar) and can click on the assigned tours with student and date in the calendar 4. The user clicks on the tour and student combination that needs to be changed 5. The user is redirected to the <i>update schedule</i> page 6. The user presses the <i>remove from schedule</i> button 7. The user is redirected to the schedule page 8. The user views a message that says that the entry was successfully removed 9. The user click <i>ok</i> on the message 10. The user can view the updated schedule
Postconditions:	C) The user views the updated schedule D) The user can further navigate the application
Alternative Flows:	The tour that needs to be deleted is currently active: 3. Follow step 1-5 4. The user gets a message the tour is currently active, the user can't change any information nor delete

Garbage collection schedule

UC-Name	G1 – View garbage collection schedule of a building (Admin/superstudent)
Short informal description:	View the schedule for the garbage collection
Primary Actor:	Student/Superstudent/Admin/Syndicus (Users)
Pre-conditions:	A) The user must be logged in with a verified account B) The user is on the homepage of the application
Normal Flow:	1. The user clicks the <i>data</i> button in the menu 2. The user is redirected to a data page 3. The user selects <i>buildings</i> in a drop-down menu 4. The user gets a list of all the buildings 5. The user clicks on the building they want to see the garbage collection schedule of 6. The user is redirected to the specific building page 7. The user sees in the info a of the building the schedule for that building
Postconditions:	A) The user can further navigate the application B) The user is on the garbage collection schedule page of that building
Alternative Flows:	View the garbage schedule via the tour overview: <ol style="list-style-type: none"> Follow normal flow 1-2 The user selects <i>tours</i> in a drop-down menu The user gets a list of all the tours The user selects the tour where the building is part of The user gets redirected to the <i>tour</i> page The user gets a list of all the buildings on that specific tour Follow normal flow 5-7 View the garbage schedule via the dashboard: <ol style="list-style-type: none"> Click on the <i>dashboard</i> button The user gets an overview of all the currently active routes The user clicks on the tour the building is on The user is redirected to the <i>active tour</i> page The user clicks on the building on the tour The user is redirected to the <i>active building</i> page The user gets all the current info of that building The user is clicks on the <i>more</i> info button The user is redirected to the specific building page The user sees in the info a of the building the schedule for that building

UC-Name	G2 – View garbage collection of a building (Syndic)
Short informal description:	The syndicus view the garbage collection of an owned building of his
Primary Actor:	Syndic
Pre-conditions:	A) The user is logged in with a verified account B) The user is on the homepage of the application
Normal Flow:	1. The syndic clicks the <i>data</i> button in the menu 2. The syndic is redirected to a data page

	<ol style="list-style-type: none"> The syndic gets an overview of all his owned buildings The syndic clicks on the building they want to see the garbage collection schedule of The user is redirected to the specific building page The user sees in the info a of the building the schedule for that building
Postconditions:	<ol style="list-style-type: none"> The syndic can further navigate the application The user is on the garbage collection schedule page of that building
Alternative Flows:	/

UC-Name	G3 – Add garbage collection info to building
Short informal description:	Add a data and garbage collection type to the garbage collection schedule of a building
Primary Actor:	Syndic/superstudent/admin
Pre-conditions:	<ol style="list-style-type: none"> The user has a verified account and is logged in The user is on the garbage collection schedule of the building where the info needs to be added (see G1/ G2)
Normal Flow:	<ol style="list-style-type: none"> The user clicks on the button <i>add to schedule</i> The user is redirected to the <i>add to garbage schedule</i> page The user fills in the required fields (i.e., garbage type and date) The user clicks on the <i>add</i> button The user is redirected back to the updated garbage collection schedule of that building
Postconditions:	<ol style="list-style-type: none"> The user can further navigate the application The user is on the garbage collection schedule page of that building
Alternative Flows:	/

UC-Name	G4 – Remove garbage collection info from building
Short informal description:	Remove info from the garbage collection schedule of a building.
Primary Actor:	Syndic/superstudent/admin
Pre-conditions:	<ol style="list-style-type: none"> The user has a verified account and is logged in The user is on the garbage collection schedule of the building where the info needs to be added (see G1/ G2)
Normal Flow:	<ol style="list-style-type: none"> The user selects the part of the info of the schedule that needs to be removed The user clicks the <i>remove</i> button of that specific info The user is asked a confirmation of the removal. The user gets a message that the info was successfully removed. The user now sees the updated garbage collection schedule of that building.
Postconditions:	<ol style="list-style-type: none"> The user can further navigate the application The user is on the garbage collection schedule page of that building

Alternative Flows:	/
--------------------	---

UC-Name	G5 – Change garbage collection schedule of that building
Short informal description:	Change certain info from the garbage collection schedule of a building
Primary Actor:	Syndic/superstudent/admin
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the garbage collection schedule of the building where the info needs to be added (see G1/ G2)
Normal Flow:	1. The user selects the part of the info of the schedule that needs to be changed 2. The user clicks the edit button 3. The field become editable on the page and the user changes the fields. 4. The user clicks the <i>change</i> button to save the changes 5. The user now sees the updated garbage collection schedule of that building.
Postconditions:	A) The user can further navigate the application B) The user is on the garbage collection schedule page of that building
Alternative Flows:	/

UC-Name	G6 – Duplicate the garbage collection schedule of a building
Short informal description:	Duplicate info from the garbage collection schedule of a building for the next 2 weeks
Primary Actor:	Syndic/superstudent/admin
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the garbage collection schedule of the building where the info needs to be added (see G1/ G2)
Normal Flow:	1. The user clicks the <i>duplicate</i> button. 2. The user receives a confirmation that the previous 2 6. The user clicks the edit button 7. The field become editable on the page and the user changes the fields. 8. The user clicks the <i>change</i> button to save the changes 9. The user now sees the updated garbage collection schedule of that building.
Postconditions:	A) The user can further navigate the application C) The user is on the garbage collection schedule page of that building
Alternative Flows:	/

Tour

UC-Name	T1 – View tour
Short informal description:	Get the info of a tour with all the buildings on that tour
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the home page of the application
Normal Flow:	<ol style="list-style-type: none"> 1. The user clicks on the <i>data</i> button. 2. The user is redirected to the <i>data</i> page 3. The user clicks on the <i>tours button</i> 4. The user is redirected to the <i>tours</i> page where the user sees a list of all created tours 5. The user filters the tours on his preferences 6. The user clicks the tour of which he wants the info 7. The user is redirected to a page with the details of the tour, buildings on the tour
Postconditions:	A) The user can further navigate the application B) The user is on the specific tour page
Alternative Flows:	The user can view the tour (if active) via the dashboard: <ol style="list-style-type: none"> 1. The user clicks on the <i>dashboard</i> page 2. The user selects the tour he wants the info of 3. The user is redirected to a page with the details of the current active tour. 4. The user gets all the info of the tour

UC-Name	T2 – Create a new tour (superstudent/admin)
Short informal description:	Create a new tour to use
Primary Actor:	Superstudent/admin
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the home page of the application
Normal Flow:	<ol style="list-style-type: none"> 1. The user clicks the <i>data</i> page 2. The user clicks on <i>tours</i> 3. The user is redirected to the <i>tours</i> page 4. The user clicks the button <i>create new tour</i> 5. The user is redirected to a <i>create tour</i> page 6. The user adds a name, selects a region for the tour and adds the buildings he wants to add in a certain order. 7. The user clicks the <i>create</i> button 8. The user is redirected to the tours list and a message appears that the tour was successfully created.
Postconditions:	A) The user can further navigate the application B) The user is on the specific tour page
Alternative Flows:	/

UC-Name	T3 – Update a tour
Short informal description:	Update the info and buildings of a tour
Primary Actor:	Superstudent/admin

Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the home page of the application
Normal Flow:	1. The user clicks the <i>data</i> page 2. The user clicks on <i>tours</i> 3. The user is redirected to the <i>tours</i> page 4. The user selects the tour he wishes to change 5. The user is redirected to a <i>change tour</i> page 6. The user can make the changes he wishes (change name, change region, change buildings that are on the tour or change the order of buildings) 7. The user clicks <i>apply changes</i> button 8. The user is redirected to the tours list
Postconditions:	A) The user can further navigate the application B) The user is on the data page with a list of all tours
Alternative Flows:	If the tour is active (a student is currently doing the tour): 1. Follow steps 1-8 2. An error message is displayed stating that the tour is currently active, and it can be changed when the tour is completed 3. The user is redirected to the tours list

UC-Name	T4 – Remove a tour
Short informal description:	Remove a tour
Primary Actor:	Superstudent/admin
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the home page of the application
Normal Flow:	1. The user clicks the <i>data</i> page 2. The user clicks on <i>tours</i> 3. The user is redirected to the <i>tours</i> page 4. The user selects the tour he wishes to remove and clicks on the <i>remove</i> button 5. The user gets a confirmation warning of the removal. 6. The user sees the updated list of tours
Postconditions:	A) The user can further navigate the application B) The user is on the data page with a list of all tours
Alternative Flows:	/

Mail

UC-Name	M1 – View a mail template
Short informal description:	View an already created mail template
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the home page of the application
Normal Flow:	1. The user clicks the <i>communication</i> tab 1. The user gets redirected to a page where all the mail templates are displayed 2. The user selects the templates he wants to see 3. The user is redirected to an info view of the template with details
Postconditions:	A) The user can further navigate the application B) The user is on the specific template page
Alternative Flows:	/

UC-Name	M2 - Create a new mail template (Admin/superstudent)
Short informal description:	Creates a new mail template that can be used to check mail
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the home page of the application
Normal Flow:	1. The user clicks the <i>communication</i> tab 2. The user gets redirected to a page where all the mail templates are displayed 3. The user clicks the button <i>create a new mail template</i> 4. The user is redirected to a form for the mail template 5. The user creates a mail template and gives it a mail 6. The user clicks the save button 7. The user is redirected to a list with all templates and a message that the template was successfully saved
Postconditions:	A) The user can further navigate the application B) The user is on the specific template page
Alternative Flows:	/

UC-Name	M3 – Change an existing mail template (Admin/superstudent)
Short informal description:	Change an existing mail template
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user has a verified account and is logged in A) The user is on the home page of the application
Normal Flow:	1. The user clicks the <i>communication</i> tab 2. The user gets redirected to a page where all the mail templates are displayed 3. The user clicks the <i>edit</i> button of a mail template that he wants to change 4. The user gets redirected to a <i>mail template update</i> page 5. The user changes the info that is displayed in the form

	6. The user clicks <i>apply changes</i> 7. The user is redirected to the overview of all templates with a message that the template was successfully updated
Postconditions:	A) The user can further navigate the application B) The user is on the communication page
Alternative Flows:	/

UC-Name	M4 - Send a mail (Admin/superstudent)
Short informal description:	Send a mail using an email template
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the template page of the mail he wants to send
Normal Flow:	1. The user clicks on the <i>send mail</i> button. The user is redirected to a form page. 2. The user selects the receiver of the mail: either typing it manually or selecting from a list of all the users of the application 3. The user clicks on <i>send email to send the mail</i> 4. The user gets a confirmation that the mail was successfully sent
Postconditions:	A) The user can further navigate the application B) The user is on the communication page
Alternative Flows:	The user wants to send an email with attachment of pictures from a building on a tour: 1. The user clicks on the data. 2. The user is redirected to the data page. 3. The user selects building. 4. The user selects the building of which he wants to send the attachments from the list. 5. The user is redirected to the <i>building specific</i> page. 6. The user is clicking the <i>view building data of tours</i> button. 7. The user is redirected to a page with a list of all the dates this building was on a tour. 8. The user selects the date he needs from the list 9. The user is redirected to a page with all the info and pictures of the building on that date. 10. The user can select the images he wants to add as attachment. 11. The user clicks the add to mail button 12. The user is redirected to a list of all templates 13. The user selects the template he needs. 14. Normal flow 2-4

UC-Name	M5 – Remove a mail template (Admin/superstudent)
Short informal description:	Remove a mail template from the list of mail templates
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user has a verified account and is logged in

	B) The user is on the template page of the mail he wants to remove
Normal Flow:	<ol style="list-style-type: none"> 1. Click the <i>remove</i> button 2. The system asks the user for confirmation of the removal 3. The user clicks <i>remove</i> 4. The user is redirected to the list of mail templates page
Postconditions:	<p>A) The user can further navigate the application</p> <p>B) The user is now on the list of templates view</p>
Alternative Flows:	/

Analysis

UC-Name	A1 – View analysis of a student
Short informal description:	View the analysis of a specific student, including total time worked in a week and how long it took to complete each building
Primary Actor:	Superstudent/admin
Pre-conditions:	A) The user is on the welcome page B) The user has an account and is currently logged in
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Analysis</i> 2. The user is redirected to the analysis dashboard 3. Click on the <i>Students</i> icon 4. The user is redirected to the analysis overview of all students 5. The user can view all students that worked this week 6. Click on the student you want to analyse 7. The user is redirected to the analysis of the specified student 8. The user can see the full analysis of the student, including the total time worked and how long it took to complete each building
Postconditions:	A) The user can see the full analysis of the student
Alternative Flows:	You want to filter the overview on specific parameters <ol style="list-style-type: none"> 1. Normal flow steps 1-5 2. On the left, specify on the multiselect menus on what parameters to filter. Options are Student, Tour, Week and Building 3. The database is queried with these parameters 4. The user can now see the filtered outcome 5. Jumps to Normal flow step 6

UC-Name	A2 – View analysis of a building
Short informal description:	View the analysis of a specific building, including how long it took to complete it in each tour
Primary Actor:	Superstudent/admin
Pre-conditions:	A) The user is on the welcome page B) The user has an account and is currently logged in
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Analysis</i> 2. The user is redirected to the analysis dashboard 3. Click on the <i>Buildings</i> icon 4. The user is redirected to the analysis overview of all buildings 5. The user can view all buildings 6. Click on the building you want to analyse 7. The user is redirected to the analysis of the specified building 8. The user can see the full analysis of the building, including how long it took to complete it in each tour
Postconditions:	A) The user can see the full analysis of the building

Alternative Flows:	<p>You want to filter the overview on specific parameters</p> <ol style="list-style-type: none"> 1. Normal flow steps 1-5 2. On the left, specify on the multiselect menus on what parameters to filter. Options are Week and Building 3. The database is queried with these parameters 4. The user can now see the filtered outcome 5. Jumps to Normal flow step 6
--------------------	--

UC-Name	A3 – View analysis of a tour
Short informal description:	View the analysis of a specific tour, including each building, which student was responsible, the total time it took to complete as well as the average time it takes to complete
Primary Actor:	Superstudent/admin
Pre-conditions:	<ol style="list-style-type: none"> A) The user is on the welcome page B) The user has an account and is currently logged in
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Analysis</i> 2. The user is redirected to the analysis dashboard 3. Click on the <i>Tours</i> icon 4. The user is redirected to the analysis overview of all tours 5. The user can view all tours 6. Click on the tour you want to analyse 7. The user is redirected to the analysis of the specified tour 8. The user can see the full analysis of the tour, including how long it took to complete each building, as well as the average time it takes to complete
Postconditions:	A) The user can see the full analysis of the tour
Alternative Flows:	<p>You want to filter the overview on specific parameters</p> <ol style="list-style-type: none"> 1. Normal flow steps 1-5 2. On the left, specify on the multiselect menus on what parameters to filter. Options are Week, Tour and Student 3. The database is queried with these parameters 4. The user can now see the filtered outcome 5. Jumps to Normal flow step 6

UC-Name	A4 – View analysis of total working hours
Short informal description:	View the analysis of the total hours worked in a specific week. This includes a listing of all students, together with how long they worked, how long they were expected to work, and for which tours they were responsible.
Primary Actor:	Superstudent/admin
Pre-conditions:	<ol style="list-style-type: none"> A) The user is on the welcome page B) The user has an account and is currently logged in
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Analysis</i> 2. The user is redirected to the analysis dashboard 3. Click on the <i>Working hours</i> icon 4. The user is redirected to the analysis overview of the working hours

	5. The user can view the total hours worked in the current week. This view is sorted based on the students that worked this week.
Postconditions:	A) The user can see the full analysis of the total hours worked in a certain week
Alternative Flows:	<p>You want to filter the overview on specific parameters</p> <ol style="list-style-type: none">1. Normal flow steps 1-52. On the left, specify on the multiselect menus on what parameters to filter. Options are Week and Student3. The database is queried with these parameters4. The user can now see the filtered outcome5. Jumps to Normal flow step 5, now with a view filtered on the specific parameters

Dashboard

UC-Name	D1 – View all ongoing tours
Short informal description:	View all the tours that are currently ongoing
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user is on the welcome page B) The user has an account and is currently logged in
Normal Flow:	1. Click on <i>Dashboard</i> 2. The user is redirected to the dashboard page 3. The user can see all ongoing tours, with a progress bar, indicating the rate of completion by the responsible student
Postconditions:	A) The user is on the dashboard page B) The user sees an overview of all ongoing tours
Alternative Flows:	Not all data can be displayed on one page 1. Click on <i>Next page</i> at the bottom of the screen 2. The user's view is re-rendered, now with information that could not previously be shown The user wants to sort the view 1. Click on the <i>Sort</i> multiselect 2. Select the criteria you want to sort on 3. The user's display is now sorted There are no ongoing tours 1. A message is displayed: <i>Looks like there are no ongoing tours. Come back later.</i>

UC-Name	D2 - View all tours, filtered on certain information
Short informal description:	View all the tours (both ongoing and past tours). This information can be filtered on Week, Student, Region, and specific Tour
Primary Actor:	Admin/superstudent
Pre-conditions:	C) The user is on the welcome page D) The user has an account and is currently logged in
Normal Flow:	4. Click on <i>Dashboard</i> 5. The user is redirected to the dashboard page 6. The user can see all ongoing tours, with a progress bar, indicating the rate of completion by the responsible student 7. Click on the <i>Week</i> select to specify a week 8. Click on the <i>Student</i> multiselect to specify a student 9. Click on the <i>Tour</i> multiselect to specify a tour 10. Click on the <i>Region</i> multiselect to specify a region 11. The database is queried on the specific parameters the user provided 12. The user now sees all the data that confirms with the specified parameters
Postconditions:	C) The user is on the dashboard page D) The user sees an overview of all tours that confirm with the given parameters

Alternative Flows:	<p>Not all data can be displayed on one page</p> <ol style="list-style-type: none"> 1. Click on <i>Next page</i> at the bottom of the screen 2. The user's view is re-rendered, now with information that could not previously be shown <p>The user wants to sort the view</p> <ol style="list-style-type: none"> 1. Click on the <i>Sort</i> multiselect 2. Select the criteria you want to sort on 3. The user's display is now sorted <p>There are no tours that confirm with the given parameters</p> <ol style="list-style-type: none"> 1. A message is shown: <i>Could not find such tours</i>
--------------------	--

UC-Name	D3 – View specific information about a tour
Short informal description:	View the specific information about a tour. This includes all buildings, the responsible student, the date, the status of each building, the time it took to complete this building, photos and notes
Primary Actor:	Admin/superstudent
Pre-conditions:	<ol style="list-style-type: none"> A) The user is on the welcome page B) The user has an existing account and is currently logged in
Normal Flow:	<ol style="list-style-type: none"> 1. Click on <i>Dashboard</i> 2. The user is redirected to the dashboard page 3. Find the tour you want specific information about (see flow D2: 6-12) 4. Click on the name of the tour 5. The user is redirected to the specific tour page 6. The user sees a display of all specific information
Postconditions:	A) The user can see all specific information about a tour
Alternative Flows:	/

UC-Name	D4 – View specific information about a building on a tour
Short informal description:	View the specific information about a building on a tour. This includes the name and owner, all taken pictures, all remarks made by the student, the manual
Primary Actor:	Admin/superstudent
Pre-conditions:	<ol style="list-style-type: none"> C) The user is on the welcome page D) The user has an existing account and is currently logged in
Normal Flow:	<ol style="list-style-type: none"> 1. Follow flow D3 to find the specific tour 2. The user can see all specific information about a tour 3. Click on the building of which you want to see the information 4. The user is redirected to the specific information page of the building 5. The user sees a display of all specific information
Postconditions:	B) The user can see all specific information about a building on a tour
Alternative Flows:	/

UC-Name	D5 – View tours planned this week (Student)
Short informal description:	View the tours that the student is assigned to
Primary Actor:	Student
Pre-conditions:	A) The user is on the welcome page B) The user has an existing account and is currently logged in
Normal Flow:	1. The user clicks <i>dashboard</i> in the menu 2. The user is redirected to the dashboard page 3. The user sees a list of all planned tours under <i>this week</i>
Postconditions:	1) The user is on the main dashboard page 2) The user can further navigate the application
Alternative Flows:	/

UC-Name	D6 – View information about previous tours (Student)
Short informal description:	View the tour that a student has done in the past
Primary Actor:	Student
Pre-conditions:	A) The user is on the dashboard page B) The user has an existing account and is currently logged in
Normal Flow:	1. The user clicks on the <i>view previous tours</i> button 2. The user is redirected to a new page <i>previous tour</i> 3. The user gets an overview of all the tours he has done in the past
Postconditions:	A) The user can view a specific finished tour by clicking on it B) The user is on the tour overview page C) The user can further navigate the application
Alternative Flows:	/

UC-Name	D7 – Start a tour (Student)
Short informal description:	The user starts a tour on the day he was assigned
Primary Actor:	Student
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the dashboard page of the application C) It is the current day the task is assigned on the schedule
Normal Flow:	1. The student is on the dashboard page of the application 2. The student clicks on the assigned task under the <i>today</i> header 3. The student is redirected to an overview of the assigned tour, with the buildings in order 4. The student clicks <i>start tour</i> button 5. The student is redirected to the first building on the tour where he is expected to upload a photo on arrival, of the garbage and a photo when leaving
Postconditions:	A) The student can further do the rest of the tour
Alternative Flows:	/

UC-Name	D8 – Do a building on a tour (Student)
Short informal description:	Once a student has started doing a tour, he needs to do each building on the tour
Primary Actor:	Student
Pre-conditions:	A) The user has a verified account and is logged in B) The student has followed use case D7 (start a tour)
Normal Flow:	<ol style="list-style-type: none"> 1. The student gets a form view of the building on the tour. 2. Once the student has arrived at the building, he uploads his first photo of <i>arrival</i> in the form (can be multiple) 3. The student uploads a second picture of the garbage containers in the form (can be multiple) 4. The student uploads a last picture of when leaving the building to the form. 5. The student can also upload comments with picture to the form (as many as needed). 6. The student clicks on <i>Go to next building</i> 7. The student is redirected to an overview and form of the next building on the tour
Postconditions:	A) The student has successfully completed a building on a tour B) The student can continue the rest of the tour
Alternative Flows:	<p>Not all pictures (step 2-4 in normal flow) are uploaded when clicking the <i>go to the next building</i>:</p> <ol style="list-style-type: none"> 1. The student gets an error message that not all pictures are uploaded and this needs to be the case in order to go to the next building. <p>This is the last building on the tour:</p> <ol style="list-style-type: none"> 1. The student is displayed a button <i>Finish tour</i> instead of <i>Go to next building</i> 2. The student clicks the button 3. The student is redirected to a final overview with all the taken pictures of each building and comments 4. The student clicks the button <i>Complete the tour</i> 5. The student is redirected back to the dashboard, a message displayed that the tour was successfully completed.

Building

UC-Name	B1 – View a building (Admin/Superstudent)
Short informal description:	View the info page of a building
Primary Actor:	Admin/Superstudent
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the home page
Normal Flow:	1. Click the <i>data</i> tab from the menu 2. The user is redirected to the data overview 3. The user select <i>building</i> from the selection option 4. A list with buildings is displayed 5. The user clicks on the building he wants the info of 6. The user is redirected to a detailed info page of the selected building
Postconditions:	A) The user is now on the specific building page B) The user can further navigate the application
Alternative Flows:	Select a building from the tour tab: 1. Follow step 1-2 in Normal flow 2. The user selects <i>tour</i> from the selection option 3. A list with tour is displayed 4. The user selects a tour where the building is part of 5. The user gets is redirected to a tour info page 6. 5-6 in Normal Flow

UC-Name	B2 – View a building (Syndic)
Short informal description:	View the info page of a building
Primary Actor:	Syndic
Pre-conditions:	C) The user has a verified account and is logged in D) The user is on the dashboard page
Normal Flow:	1. The user gets a list of all the buildings he owns 2. The user clicks on the building of which he wants the info to see 3. The user is redirected to a detailed info page of the selected building
Postconditions:	A) The user is now on the specific building page B) The user can further navigate the application
Alternative Flows:	/

UC-Name	B3 – Add a building (Admin/Syndic)
Short informal description:	Add a building to the list of buildings
Primary Actor:	Admin/superstudent
Pre-conditions:	A) The user has a verified account and is logged in B) The user is on the dashboard page
Normal Flow:	1. The user clicks on the <i>data</i> tab in the menu 2. The user is redirected to the data page 3. The user clicks <i>building</i> from the menu 4. The user sees a list of all buildings

	<ol style="list-style-type: none"> The user clicks the <i>add a building</i> button The user is redirected to a form for a new building The user fills in the form and selects a syndic The user clicks the button <i>add building</i> The user is redirected to the building list, a message appears that the building was successfully created
Postconditions:	<ol style="list-style-type: none"> The user is now on the building data page The user can further navigate the application
Alternative Flows:	/

UC-Name	B4 – Remove a building (Admin/superstudent)
Short informal description:	Remove a building from the building list
Primary Actor:	Admin/superstudent
Pre-conditions:	<ol style="list-style-type: none"> The user is successfully logged in with a verified account The user is on the specific building info page
Normal Flow:	<ol style="list-style-type: none"> The user clicks on the <i>remove</i> button The user receives a confirmation message stating that it will also remove the building from all the tours The user clicks <i>ok</i> The user is redirected to the building page
Postconditions:	<ol style="list-style-type: none"> The user is now on the building data page The user can further navigate the application
Alternative Flows:	<p>The building is currently active on a tour:</p> <ol style="list-style-type: none"> Normal flow 1-3 A warning is displayed stating that the building can't be removed as it is active on a tour Normal flow step 4

UC-Name	B5 – Change a building (superstudent/admin)
Short informal description:	Change a building from the building list
Primary Actor:	Superstudent/admin
Pre-conditions:	<ol style="list-style-type: none"> The user is successfully logged in with a verified account The user is on the data building page (Follow B3 step 1-4)
Normal Flow:	<ol style="list-style-type: none"> Select the pencil icon of the building you want to change The user is redirected to a form with all the info about the building filled in. The user changes the form The user clicks save changes The user gets a confirmation warning with that the changes will be applied The user clicks <i>ok</i> The user is redirected back to the data building page
Postconditions:	<ol style="list-style-type: none"> The user is now on the building data page The user can further navigate the application
Alternative Flows:	<p>The building is currently active on a tour:</p> <ol style="list-style-type: none"> Normal flow 1-4

	<ol style="list-style-type: none"> 2. A warning is displayed stating that the building can't be changed as it is active on a tour 3. Normal flow step 7
--	---

UC-Name	B6 – Upload a manual for a building (Syndic/admin/superstudent)
Short informal description:	Upload a manual for a building
Primary Actor:	Syndic/admin/superstudent
Pre-conditions:	<ol style="list-style-type: none"> A) The syndic is logged in with a verified account B) The syndic is on the specific page of the application
Normal Flow:	<ol style="list-style-type: none"> 1. The user clicks on the add new manual button 2. This opens a pane where you can select/drop a manual file 3. The user adds a manual 4. The user clicks <i>save new file</i> 5. The file is uploaded to the server 6. The pane closes, the manual can be viewed in the manuals tab of the building
Postconditions:	<ol style="list-style-type: none"> A) The user is now on the building info page B) The user can further navigate the application
Alternative Flows:	/