

Group 30

Trainer+

ECS506U Software Engineering Group Project

Requirements Elicitation Report

1. List of Requirements

| ID | Requirement | Type | Priority | Use Case |
|------------------|---|----------------|----------|-------------------------------------|
| <i>All Users</i> | | | | |
| 1 | Trainer+ must allow all users to have a unique username. | Functional (D) | Core | Log in |
| 2 | All users must have a password. | Functional (D) | Core | Log in |
| 3 | If a user doesn't have a username, they will be allowed to create one by using their FDM issued ID. | Functional (D) | Core | Register |
| 4 | All users must be able to reset their password if lost or forgotten. | Functional | Core | Reset Password |
| 5 | A user must be allowed to add an email address. | Functional (D) | Core | Register (or Edit Personal Details) |
| 6 | They can add a phone number. | Functional (D) | Optional | Register (or Edit Personal Details) |
| 7 | All users should be able to log out of the Trainer+ app. | Functional | Core | Log out |

| Academy Scheduling Team – Course Staff | | | | |
|--|---|------------|----------|-------------------|
| 8 | Trainer+ must allow the course staff to create courses. | Functional | Core | Create Course |
| 9 | They can add name and ID to course. | Functional | Core | Create Course |
| 10 | They can add description to course. | Functional | Core | Create Course |
| 11 | They can add content/material to a course. | Functional | Optional | Create Course |
| 12 | Course staff can update a specific course. | Functional | Optional | Update Course |
| 13 | When updating a course, they can alter the course description and name. | Functional | Optional | Update Course |
| 14 | They can also change the course content. | Functional | Optional | Update Course |
| 15 | They should be able to search for a training course by ID or name. | Functional | Core | Search Course |
| 16 | They must be able to deactivate a training course. | Functional | Core | Deactivate Course |

| <i>Academy Scheduling Team – Assign Staff</i> | | | | |
|---|---|------------|------|------------------------|
| 17 | They can view a trainer's and/or trainee's calendar. | Functional | Core | Search Trainee/Trainer |
| 18 | They must be able to see what courses a trainee is meant to do. | Functional | Core | Search Trainee/Trainer |
| 19 | They must be able to see what courses a trainer can teach. | Functional | Core | Search Trainee/Trainer |
| 20 | They must view the calendar/availability for specific rooms (depending on location) to see when they're available for teaching. | Functional | Core | Search Location |
| 21 | They must be able to assign a training course to the calendar for a trainee to complete. | Functional | Core | Add Session |
| 22 | They must be able to assign a training course to the calendar for a trainer to teach. | Functional | Core | Add Session |
| 23 | They can also assign a room to a course for a specific time. | Functional | Core | Select Time Slot |
| 24 | Assign staff can also cancel a session from a trainer/trainee's calendar. | Functional | Core | Cancel Session |

| <i>Trainer & Trainee</i> | | | | |
|------------------------------|--|----------------|------|------------------------|
| 25 | Trainer can update their specialised fields/qualifications. | Functional | Core | Edit Personal Profile |
| 26 | Both trainer and trainee can view their personal details. | Functional | Core | Edit Personal Profile |
| 27 | Trainer must be able to see what fields they are specialised in. | Functional | Core | Edit Personal Profile |
| 28 | Both must be able to view their alerts for an upcoming session. | Functional | Core | View Alerts |
| 29 | Trainee is allowed to update their personal details such as phone number, email address. | Functional (D) | Core | Edit Personal Profile |
| 30 | Trainee can view the progress of the training courses assigned to them, such as date started/date completed. | Functional | Core | View Course Info |
| 31 | Both trainer and trainee can view their personal calendar. | Functional | Core | View Personal Calendar |
| 32 | Both must be able to view the courses assigned to them. | Functional | Core | View Course Info |
| 33 | Both must be able to synchronise their calendar so they don't have to be logged into Trainer+ to view their most recent synced calendar. | Functional | Core | Sync Calendar |

| Administrator | | | | |
|---------------|--|------------|------|--------------------------------------|
| 34 | They can remove a specific user from the system. | Functional | Core | Remove User |
| 35 | They can search for a user by their name, username or FDM employee ID. | Functional | Core | Search User |
| 36 | They can delete a course, that is inactive, from the system. | Functional | Core | Remove Course |
| 37 | An administrator can also change access permissions for a user in the system. | Functional | Core | Add Permission/ Remove Permission |
| 38 | They must also be able to view a trainer's profile and see their qualifications/skills. | Functional | Core | Search User |
| 39 | They will also be able to view a trainee's profile to see what training courses they are taking. | Functional | Core | Search User |
| 40 | They will be able to see each individual's data such as personal details (name, address, email). | Functional | Core | Search User |

| <i>The System</i> | | | | |
|-------------------|---|----------------|------|---|
| 41 | Trainer+ should store all the usernames and passwords in a database. | Functional | Core | - |
| 42 | The passwords must be hashed for keeping passwords safe from the developers. | Functional (D) | Core | - |
| 43 | Trainer+ must authenticate a user's login details by checking if there's a match with the database. | Functional | Core | - |
| 44 | Trainer+ should also store all courses with course ID in a database. | Functional | Core | - |
| 45 | Trainer+ must use filters to access specific data from the database, such as getting courses from one location. | Functional | Core | - |
| 46 | Trainer+ shall inform users a notification provided to inform users an update is available. | Functional | Core | - |
| 47 | Trainer+ shall provide an offline service to the user. | Functional | Core | - |

Non Functional

| ID | Requirement | Type | Priority | Use Case |
|----|--|-------------------------------|----------|----------|
| 1 | Trainer+ shall be written using Java Development Kit 8. | Non-functional Platform | Core | - |
| 2 | Trainer+ shall store data using MySQL Connector/J 8.0. | Non-functional Platform | Core | - |
| 3 | Trainer+ shall be written using JavaFX 8 for the graphical user interface. | Non-functional Platform | Core | - |
| 4 | Trainer+ shall run on Windows 10 (32/64 bit) & MacOS (64bit) operating systems. | Non-functional Portability | Core | - |
| 5 | Trainer+ shall run on personal computers (i.e., laptops & desktops) with the minimum specification requirements of 2 GB of RAM, 1 GHz of CPU, 16 GB of free hard disk space. | Non-functional Portability | Core | - |

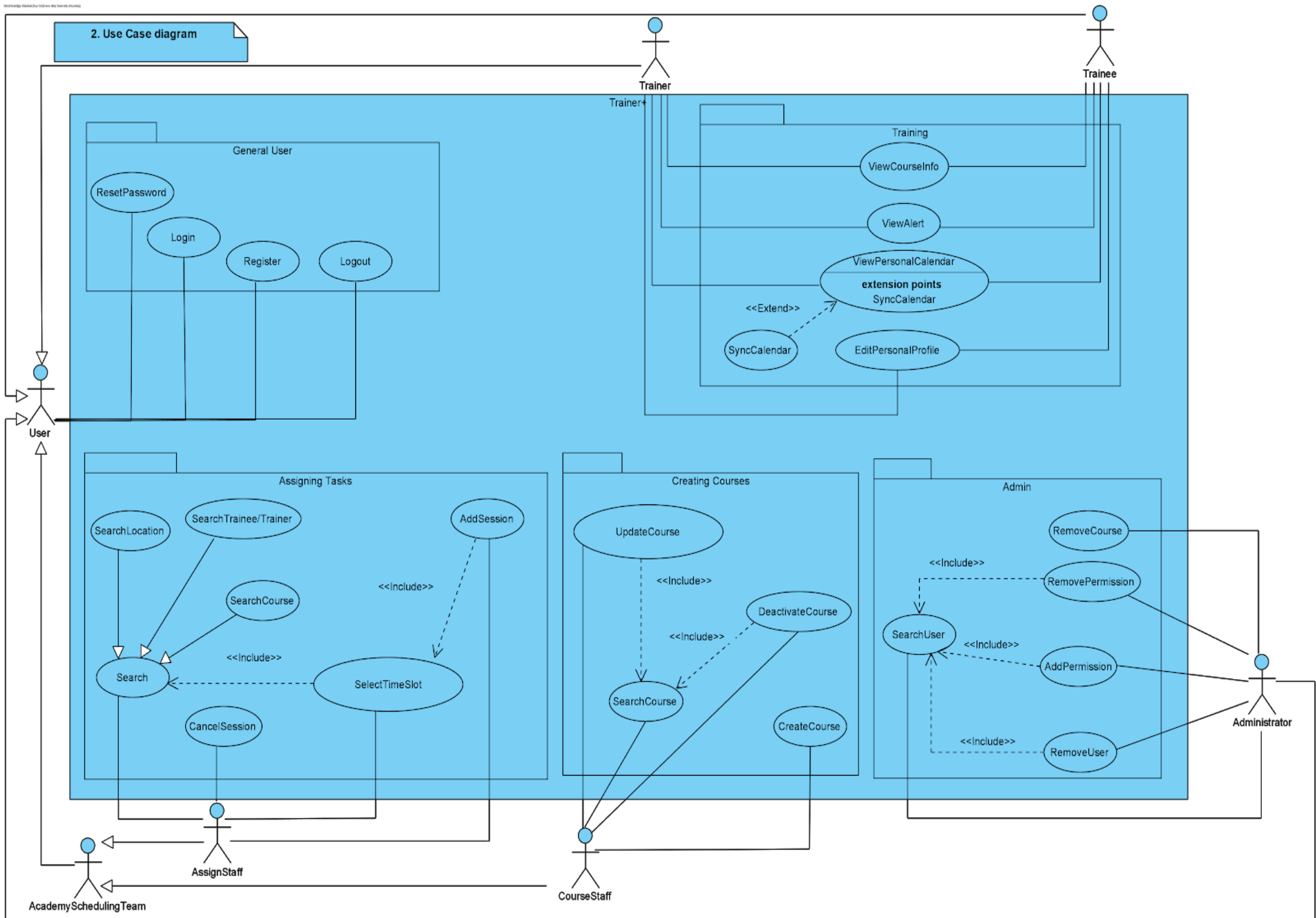
| | | | | |
|----|---|--|----------|-----------------|
| 6 | Trainer+ shall be available at least 98% uptime per year. | Non-functional Availability | Core | - |
| 7 | Trainer+ shall have a downtime of 28.80 minutes per day for maintenance between the hours of 1 am and 2 am GMT. | Non-functional Availability | Core | - |
| 8 | Trainer+ shall not allow users to perform activities within the system without authentication. | Non-functional Security | Core | - |
| 9 | Trainer+ shall protect personal data of its users in compliance with General Data Protection Regulation (GDPR) from unauthorized personnel. | Non-functional Confidentiality | Core | - |
| 10 | Trainer+ shall support a minimum of 5,000 concurrent sessions per second. | Non-functional Performance | Optional | - |
| 11 | Trainer+ shall login or logout a user within 5 seconds. | Non-functional Response Time/Speed | Core | Login Logout |

| | | | | |
|----|---|--|------|--|
| 12 | Trainer+ shall notify the trainer and trainee of an upcoming events 30 minutes before the event starts on their personal computers. | Non-functional Response Time/Speed | Core | View Alert |
| 13 | Trainer+ shall have a response time event of a maximum of 7 seconds when displaying courses, searches, the personal calendar & profile. | Non-functional Response Time/Speed | Core | View Course Info, Search Course, Search Trainee/Trainer, Search Location, Search User, View Personal Calendar, Edit Personal Profile |

| | | | | |
|----|--|--|----------|--|
| 14 | Trainer+ shall have a user input response time of a maximum of 10 seconds when creating/updating /deactivating/removing courses, altering user permissions and adding/cancelling sessions. | Non-functional Response Time/Speed | Core | Create Course Update Course Deactivate Course Remove Course Add Permission Remove Permission Add Session Cancel Session |
| 15 | Trainer+ shall occupy less than 350 megabytes of computer storage space. | Non-functional Size | Optional | - |
| 16 | Trainer+ users shall require up to 2 to 5 hours of training time to understand how to perform tasks within the system. | Non-functional Usability | Optional | - |
| 17 | Trainer+ shall store a minimum of 1000 user records in the MySQL database. | Non-functional Capacity | Core | - |

| | | | | |
|----|--|------------------------------|----------|---|
| 18 | Trainer+ shall have a maximum of 3.5% probability of data corruption on failure. | Non-functional Robustness | Optional | - |
| 19 | Trainer+ shall restart 17 to 25 minutes after failure. | Non-functional Recovery | Optional | - |
| 20 | Trainer+ shall disconnect 5% of its users if the server is using more than 92% of its resources to avoid a system failure. | Non-functional Efficiency | Core | - |

2. Use Case diagram



3. Use Case Description

Name:

- Add Session

Brief Description:

- An assign staff schedules a training session onto trainers/trainees' calendars.

Actors:

- Assign Staff

Preconditions:

- The user must be registered as an assign staff
- The assign staff must be logged in

Basic Flow: The set of steps the actors take to accomplish the goal of the use case. A clear description of what the system does in response to each user action.

1. Search and selects for the location of the training facility
 - a. System filters out all the training facilities that do not suit the assign staff's search criteria, lists out all the ones that fits the criteria allowing the user to select
2. Search and select course
 - a. System lists out all the courses at the selected training facility for the user to select
3. Select available time slots for the selected course

- a. System displays the course calendar, allowing the assign staff to select free time slots
- 4. Search and select trainers/trainees to participate in the training session
 - a. System displays all trainers that teaches, all the trainees that are enrolled in the selected course that are free at the selected time slot for the assign staff to select
- 5. Assign staff clicks 'Add session' button
 - a. System adds the training session as an event to all relevant trainers/trainees' calendar
 - b. System creates an alert of the training session for every relevant trainer/trainee

Alternate Flows:

System fails to create and add the training session onto calendar

- 5.1 System displays an error message to the assign staff displaying that the training session was not successfully created and saves the training session as a draft for easy creation in the next attempt.
- 5.2 System returns to basic flow step 5, prompt assign staff to try again.
- 5.3 If successful, add session function finish.
- 5.4 If fails again, system blocks add session functionality from assign staff, prompt critical failure and temporary unavailable.
- 5.5 System alerts administrator by sending a report displaying that there was a critical failure in saving a training session.
- 5.6 System sends the admin a troubleshooting message, displaying probable solutions.
- 5.7 System sends a detailed report to the developer.

Post Conditions:

- A training session with an existing course and valid number of trainers and trainees must be scheduled on their perspective calendars

4. Risk Assessment

| Risk | Likelihood | Severity | Impact | Preventative / Mitigating Actions |
|--------------------------------|------------|----------|--|---|
| Scope is not defined well | Low | High | Unwanted product | Ask relevant questions to the client in order to gain clear understanding of scope |
| Scope creep | Low | Medium | Unnecessary product features client didn't ask for and will not use/waste of resources | Stick to the scope described by the client and focus on the main features of the product. |
| Inaccurate time forecast | Medium | High | Delivery date push back | Plan ahead and account for unexpected events |
| Stakeholder conflict | Medium | Medium | Confusion in developing product resulting in a wrong product | Understand the needs of each different stakeholder and accommodate each of them |
| Group members miscommunicating | High | High | Unfinished product | Have regular meetings where we update each other on progress and next steps |

| | | | | |
|---|--------|--------|---|--|
| Low team motivation | Medium | Medium | Low standard product | Keep the group excited and engaged through having a shared vision and working together to finish a product |
| Lack of training for certain skills | Low | Low | Low quality product | Compile list of resources and tools we can use to increase our skill set |
| Chosen architecture not fit for purpose | Low | High | Product rendered useless | Test chosen architecture is suitable for the product |
| Design not fit for purpose | Medium | Low | Unpleasant product experience | Gather information on design preferences from user roles |
| Information security issues | High | High | Data breach | Extensively test safety measures to ensure reliability |
| Resource shortfall | Medium | High | More users than expected/planned for. Users not able to use product | Carry out sufficient research identifying our target demographic and estimating number of users |
| Health and wellness of developers | Medium | High | Insufficient number of members working therefore poor delivery of final product | Regular exercise and healthy diet. Take regular breaks and not just tied down to your desk and chair. INclude ergonomics in the work environment |

| | | | | |
|--------------------|--------|--------|---|---|
| Electrical safety | Low | Medium | Delayed delivery of product | Safe work practises when using electrical devices. Know location of shut-off/circuit breaker switch. Minimize water or chemical spill on/near electrical equipment |
| Budget risk | Low | High | Cost overruns and unforeseen project scope expansion | Identify particular factors most vulnerable to over-budgeting and develop alternative frameworks to ensure that the project cost stays within a predetermined range |
| Poor documentation | Medium | Medium | Insufficient amount of documented product knowledge to refer to | Streamline the process by establishing sensible conventions, using specialized applications to help write documentation such as Confluence and store all relevant information such that accessibility is guaranteed |