Capstone Project

SEPTEMBER 25

Kicker

Authored by: Courtney Webber

Table of Contents

Introduction

Purpose

The purpose of this web application is to assist children learning a musical instrument to set and achieve goals in their learning journey, with the guidance of their teacher. For children as well as adults, motivation is often a fleeting sensation, and this application seeks to be one part of the solution.

These days, there is an app for everything, yet not many productivity apps aimed at children. Many practise-logging applications focus on time spent, however, this doesn't necessarily result in any gains if the learner is unable to effectively utilise this time. That's where setting SMART (Specific, Measurable, Achievable, Realistic, Timely) goals come in. Kicker encourages the creation and tracking of these goals that will be created in collaboration with the students' teacher, and relies heavily on student/teacher interaction, whether face-to-face or online.

Currently, many parents are faced with paying for expensive music lessons, and can become frustrated when the child doesn't appear to take practising seriously. Furthermore, teachers are often frustrated that they can't seem to motivate practise outside of lessons and thus progress can be slow and painful. Overall, such a situation can make learning an instrument a chore for all involved, and may even drive children away from music entirely.

Ideally, Kicker will help to address this issue by providing greater interaction between students, teachers, and parents, and helping students learn how to learn by setting and achieving goals.

Industry/Domain

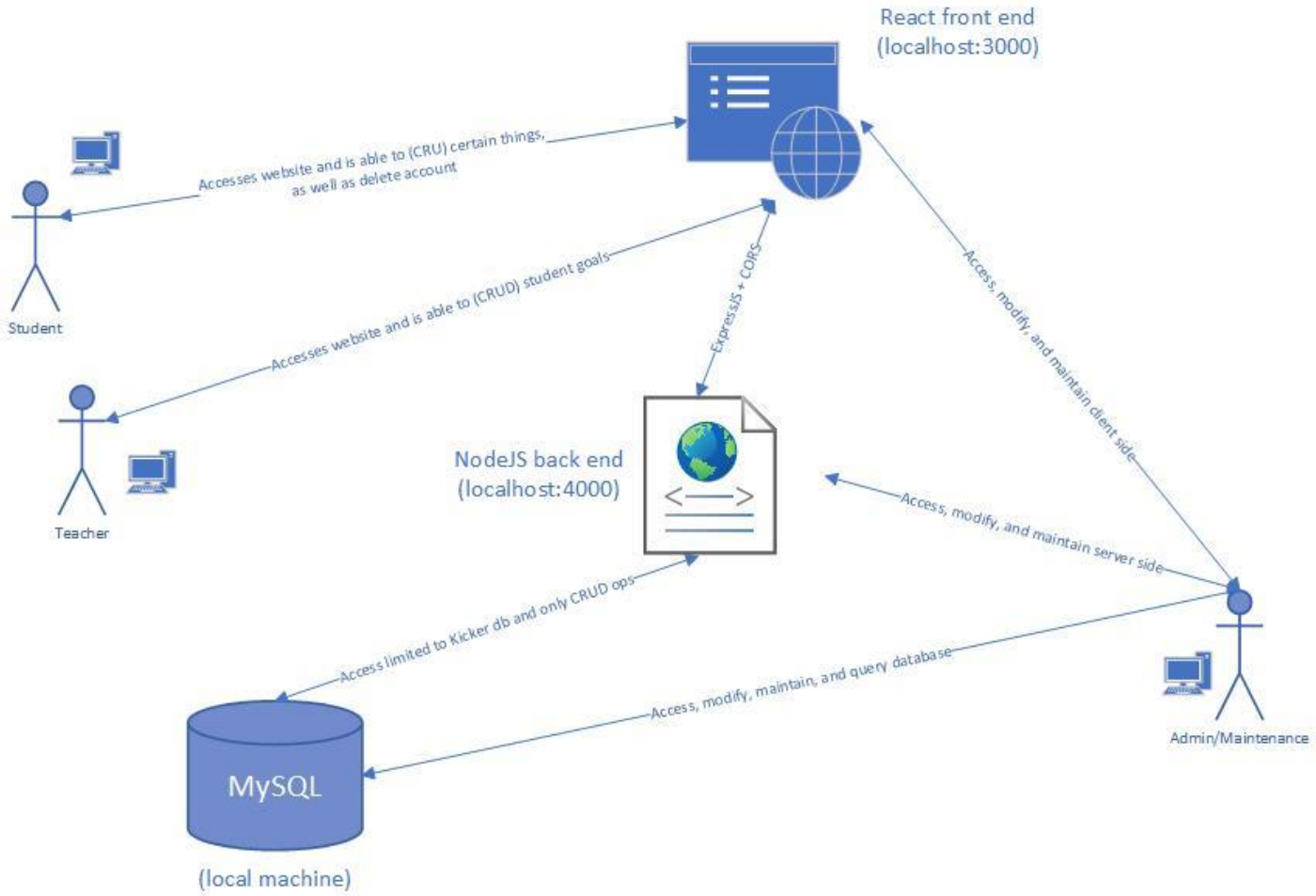
This application is designed for use in the music tuition industry. It could be relevant to other industries where students are regularly able to spend time with their teacher one-on-one, for example, private tutoring in various other subjects.

Stakeholders

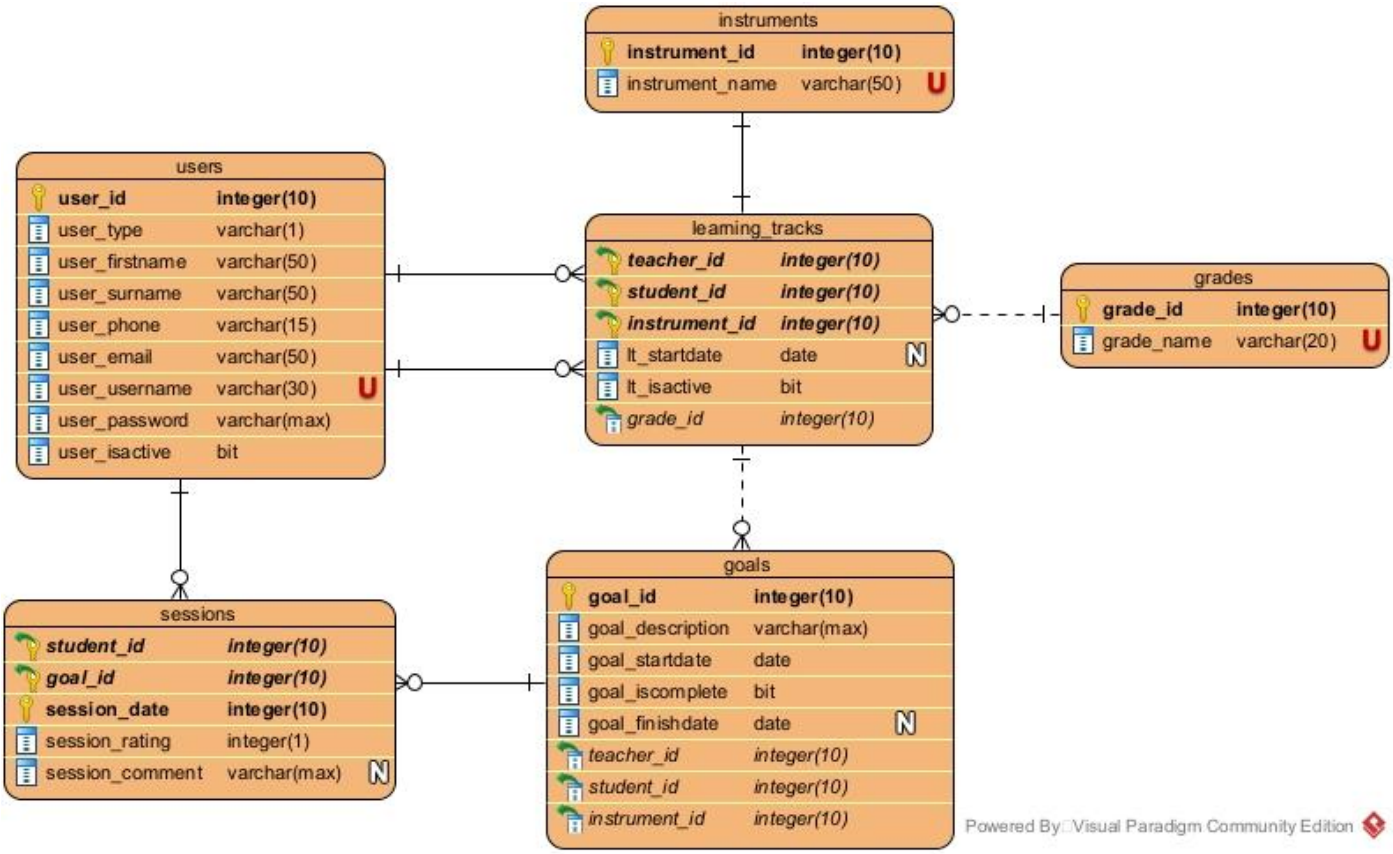
Those with access to the system are teachers and their students.
Teachers would be able to track student progress and quickly diagnose problem areas, as well as potentially seeing greater focus and improvement in their students.
Students would learn how to set different goals, assess their own strengths and weaknesses, interact more with their teachers, and recognise their progress over time. This would allow them to get a better sense of their growth, and thus help to encourage a greater sense of self-efficacy.
Stakeholders can expect a more connected and comprehensive learning experience than the traditional model of physical/static notes and little direction between lessons.

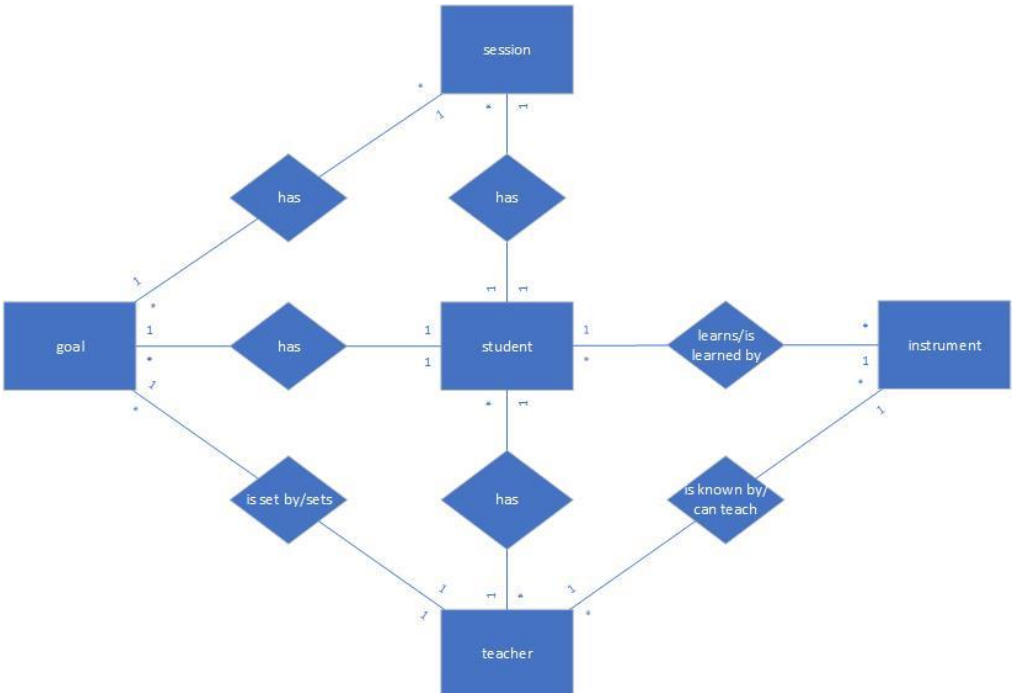
Product Description

Architecture Diagram



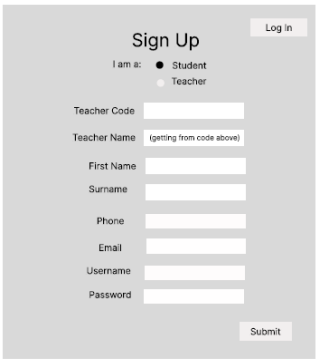
Database Diagrams

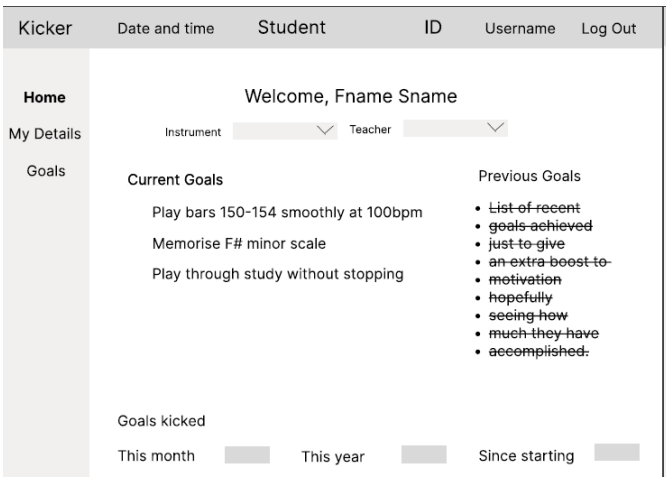


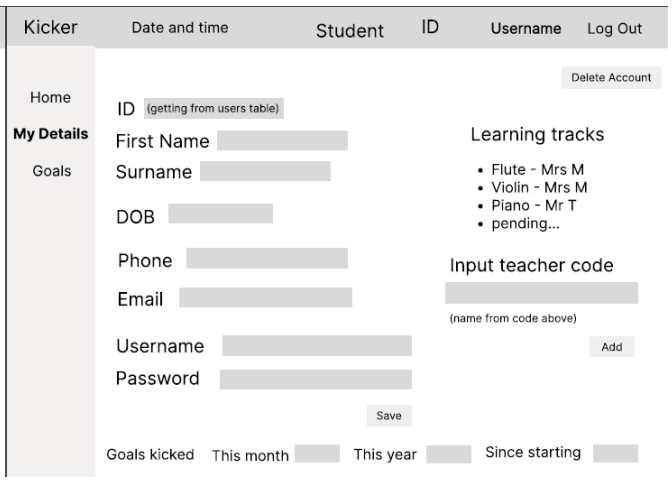


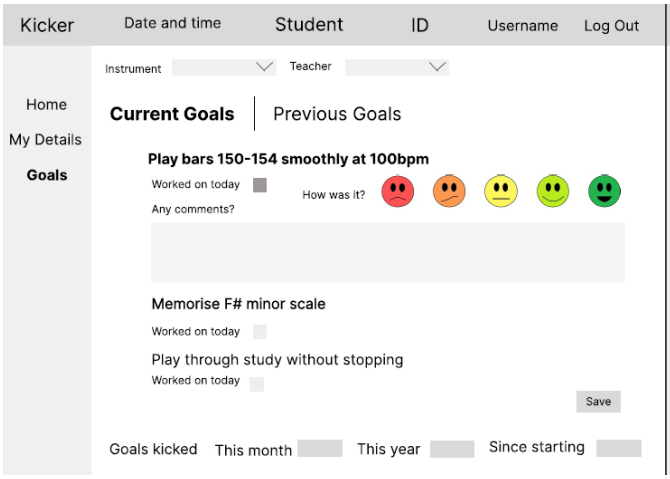
Wireframe Design

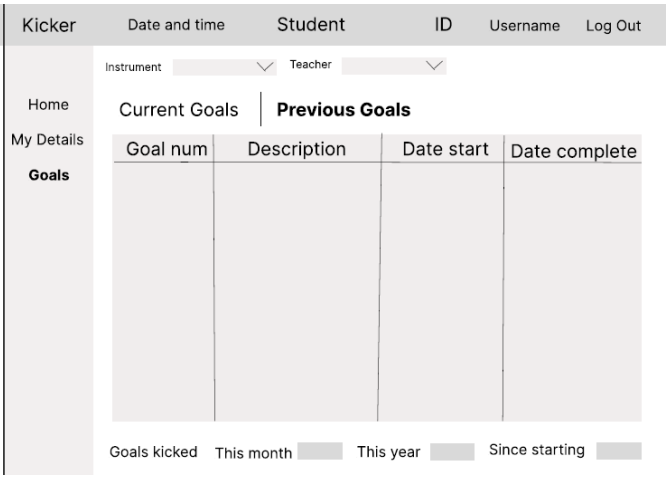
Student



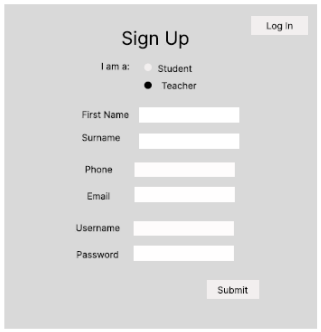


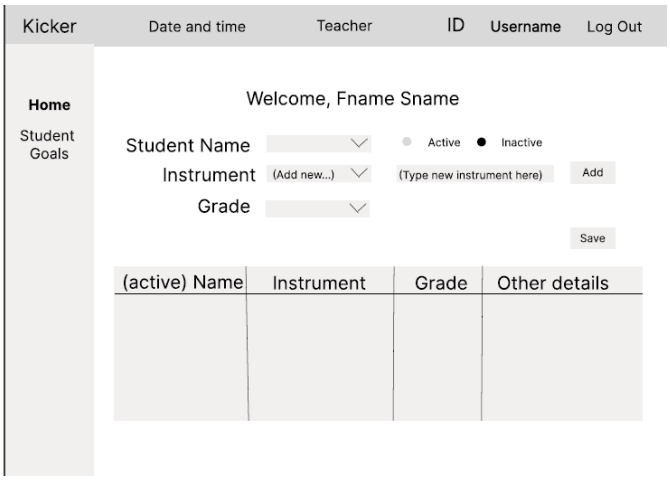


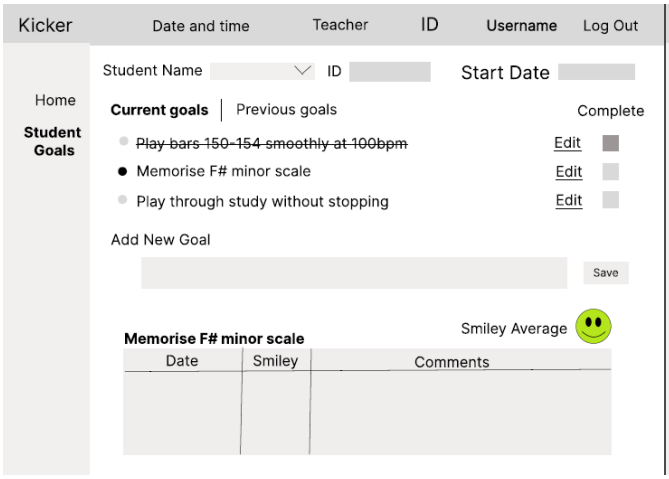


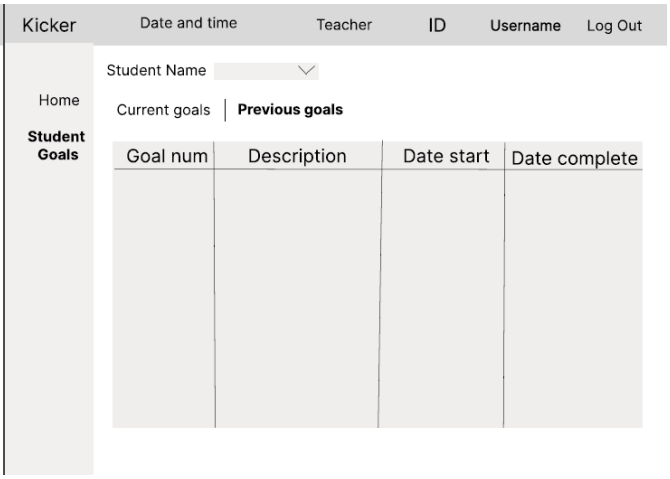


Teacher


Sign Up
Log in
I am a:
Student
Teacher
First Name
Surname
Phone
Email
Username
Password
Submit

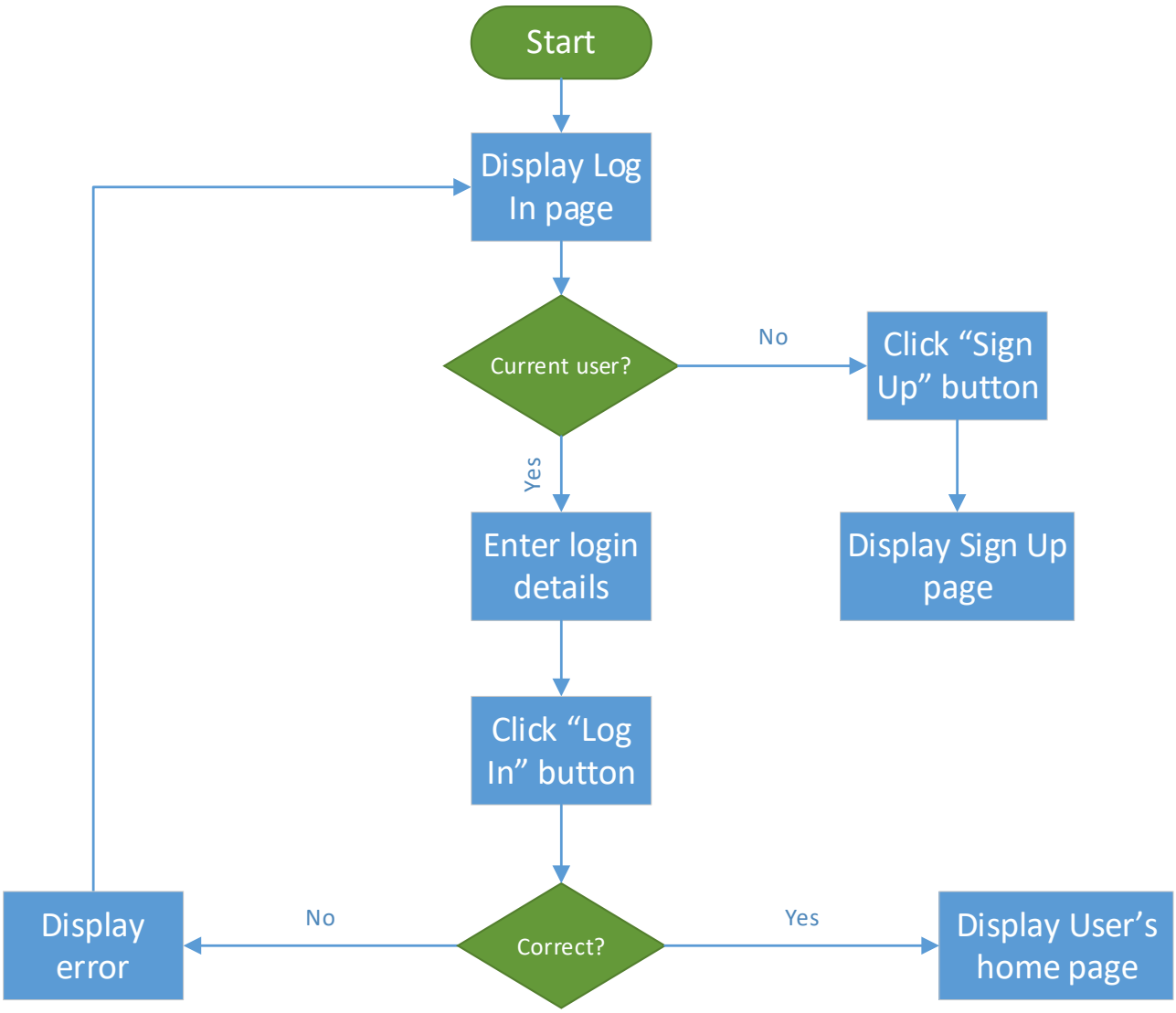

Kicker
Date and time
Teacher
ID
Username
Log Out
Home
Student Goals
Welcome, Fname Sname
Student Name
Active
Inactive
Instrument
(Add new...)
(Type new instrument here)
Add
Grade
Save
(active) Name
Instrument
Grade
Other details


Kicker
Date and time
Teacher
ID
Username
Log Out
Home
Student Goals
Student Name
ID
Start Date
Current goals
Previous goals
Complete
Play bars 150-154 smoothly at 100bpm
Edit
Memorise F# minor scale
Edit
Play through study without stopping
Edit
Add New Goal
Save
Smiley Average
Memorise F# minor scale
Date
Smiley
Comments

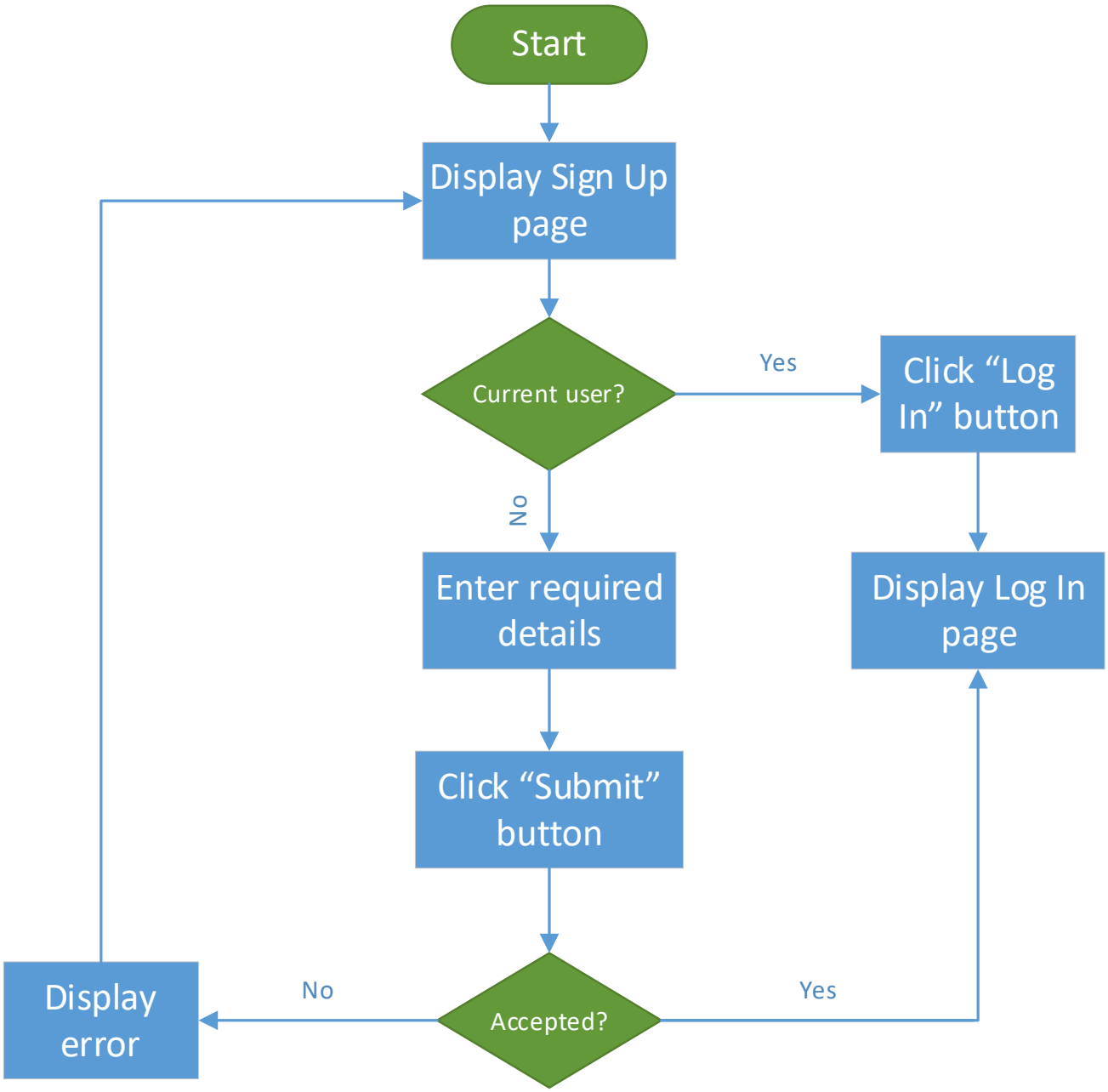

Kicker
Date and time
Teacher
ID
Username
Log Out
Home
Student Goals
Student Name
Current goals
Previous goals
Goal num
Description
Date start
Date complete

User Flow

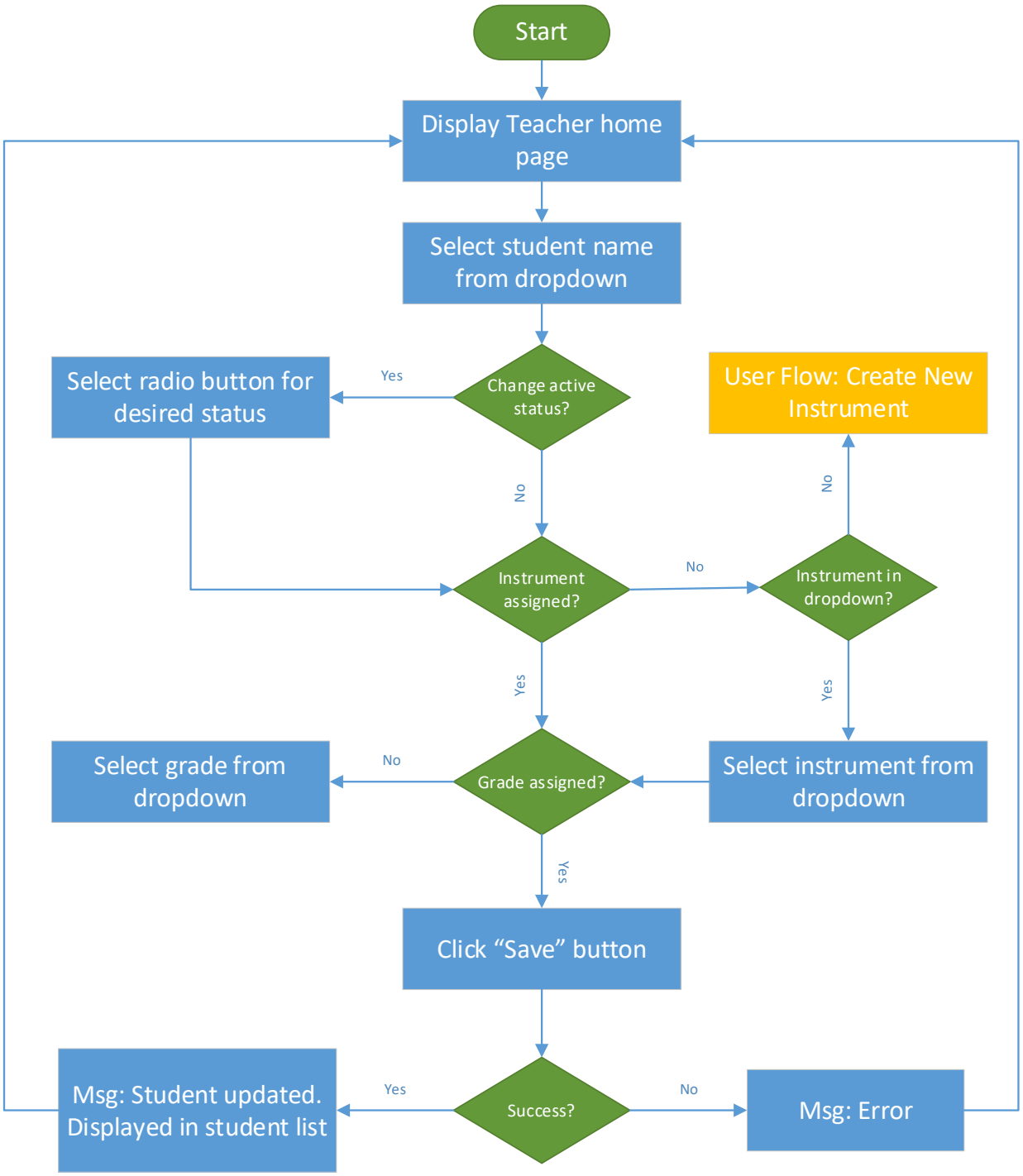
User Flow – Log In



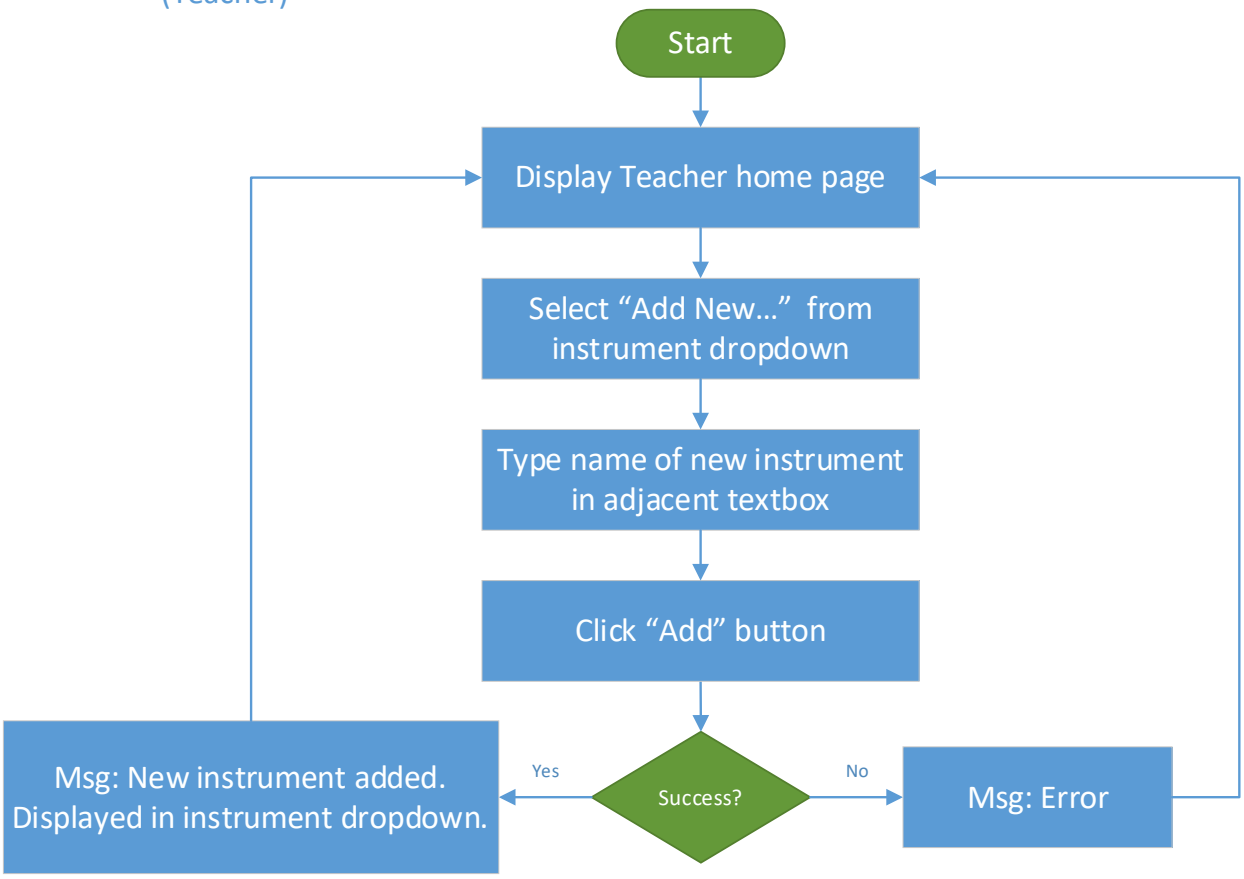
User Flow – Sign Up



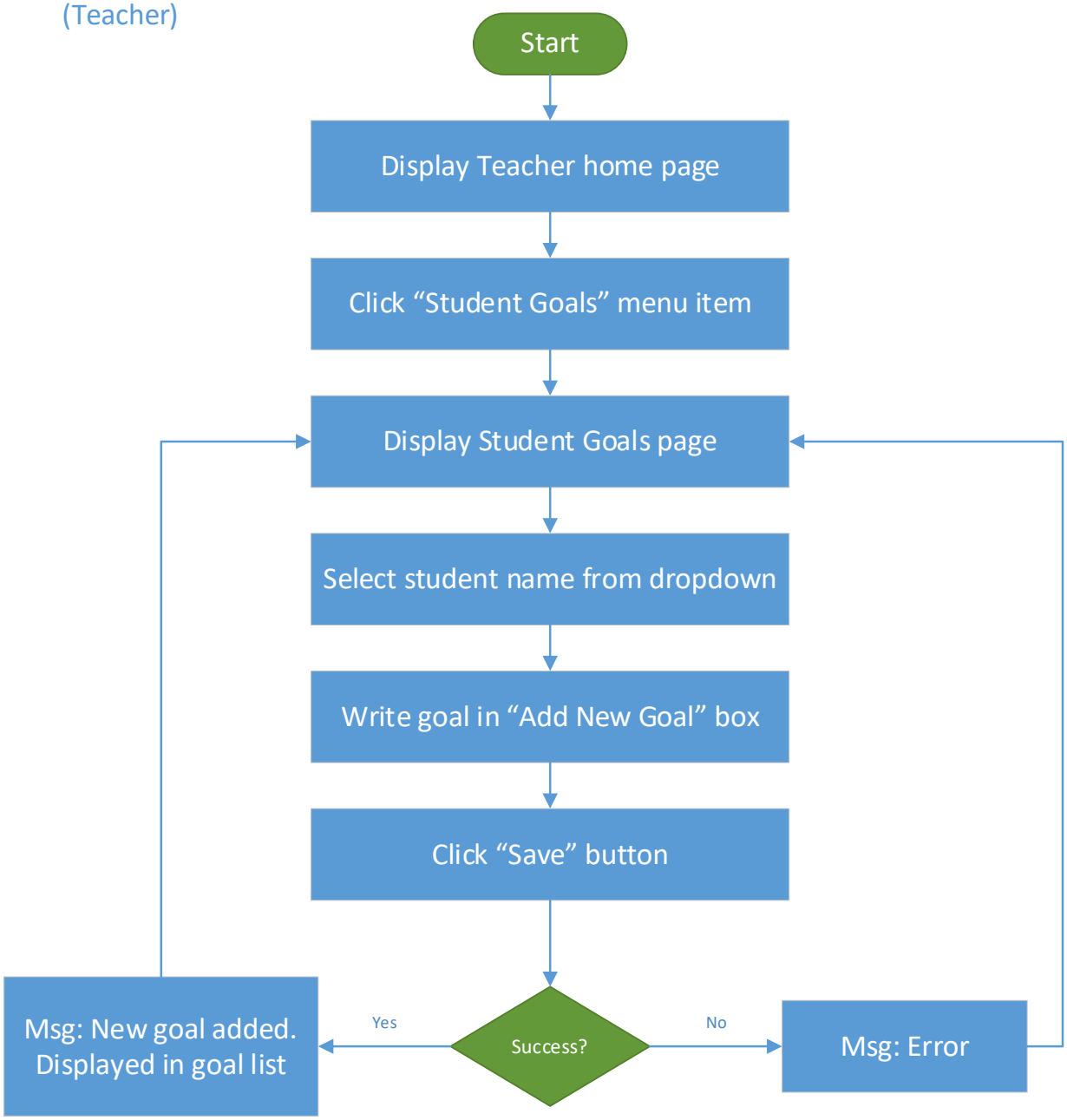
User Flow – Edit Student Details (Teacher)



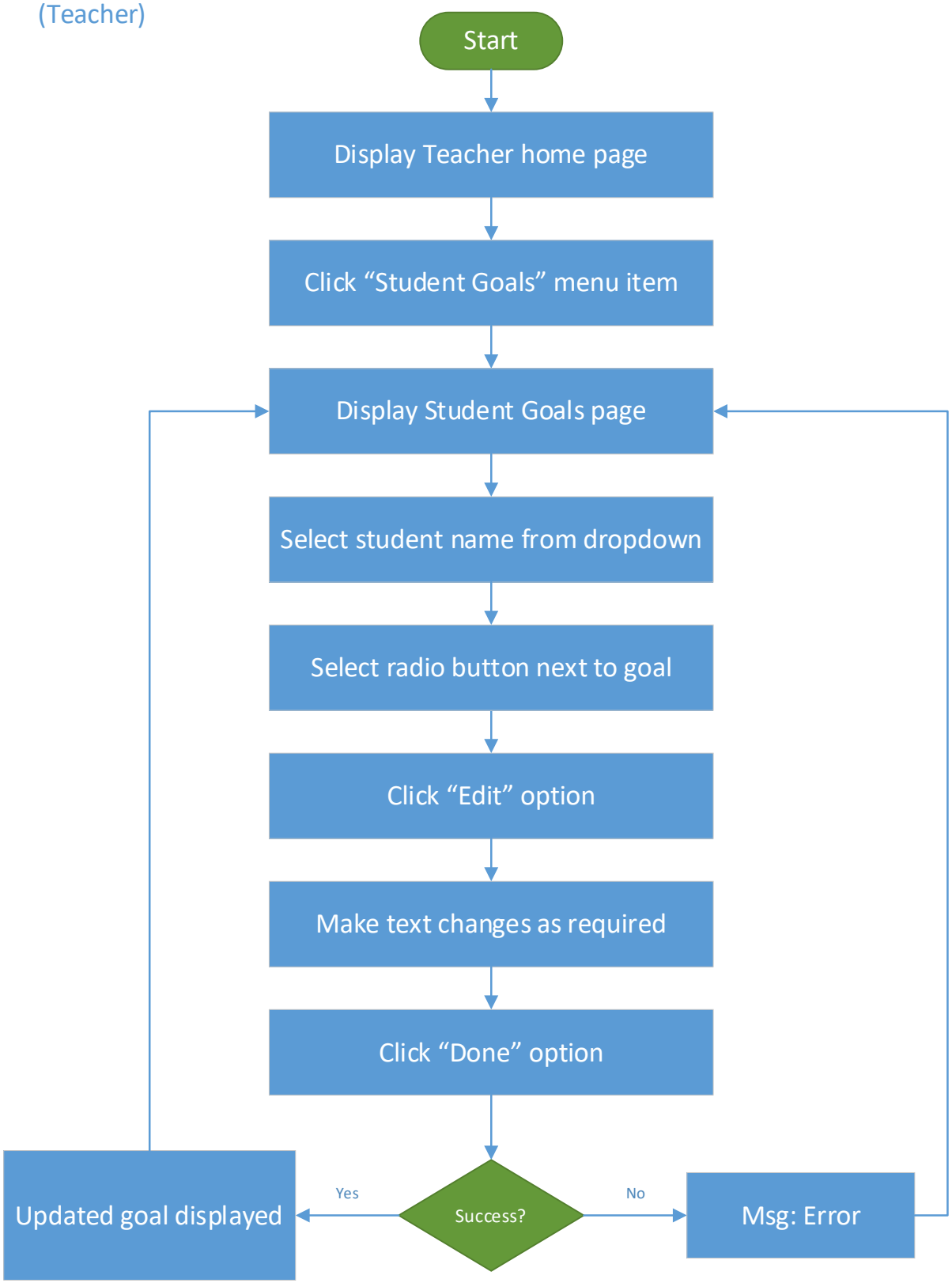
User Flow – Create New Instrument (Teacher)



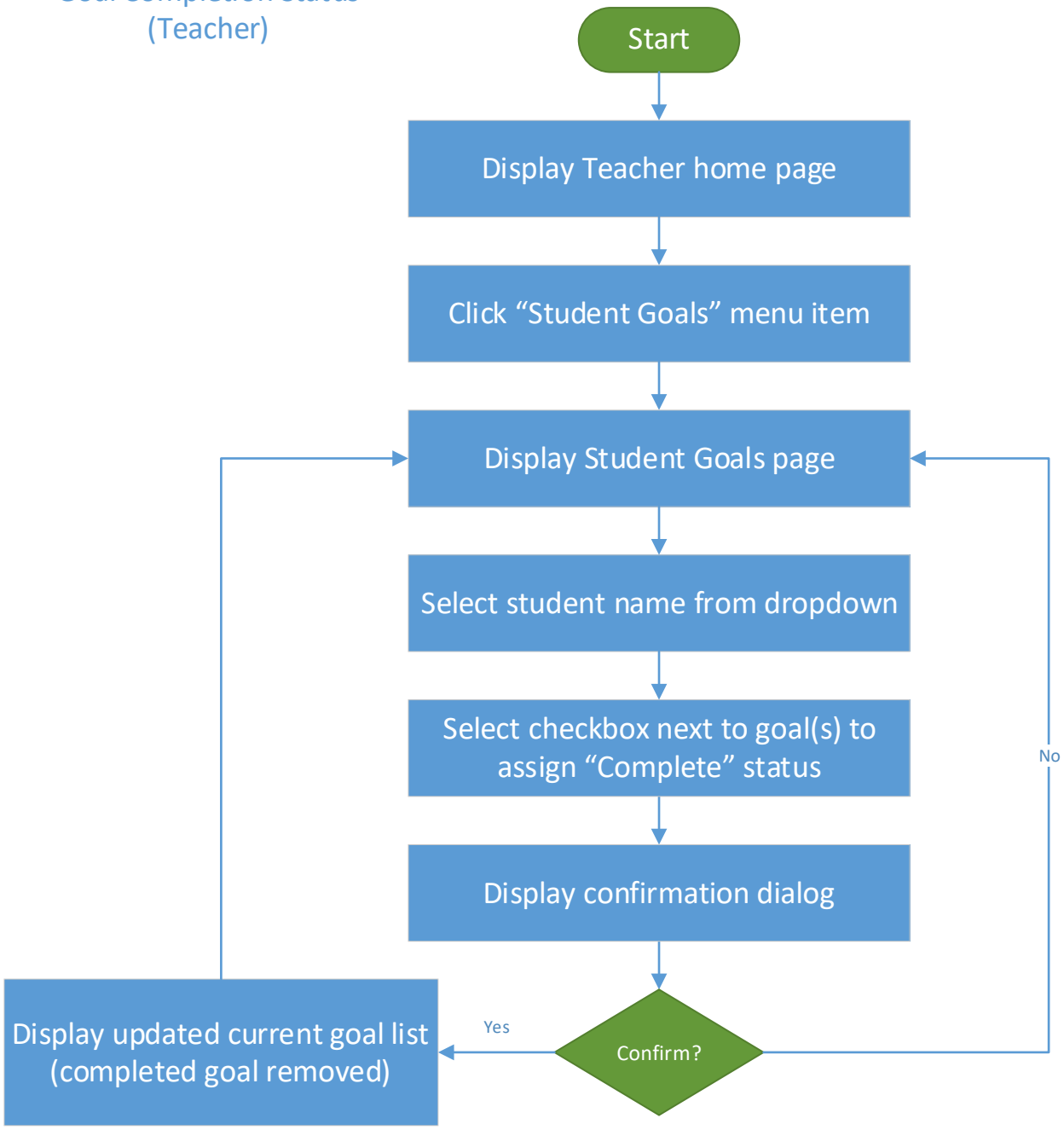
User Flow – Create Goal (Teacher)



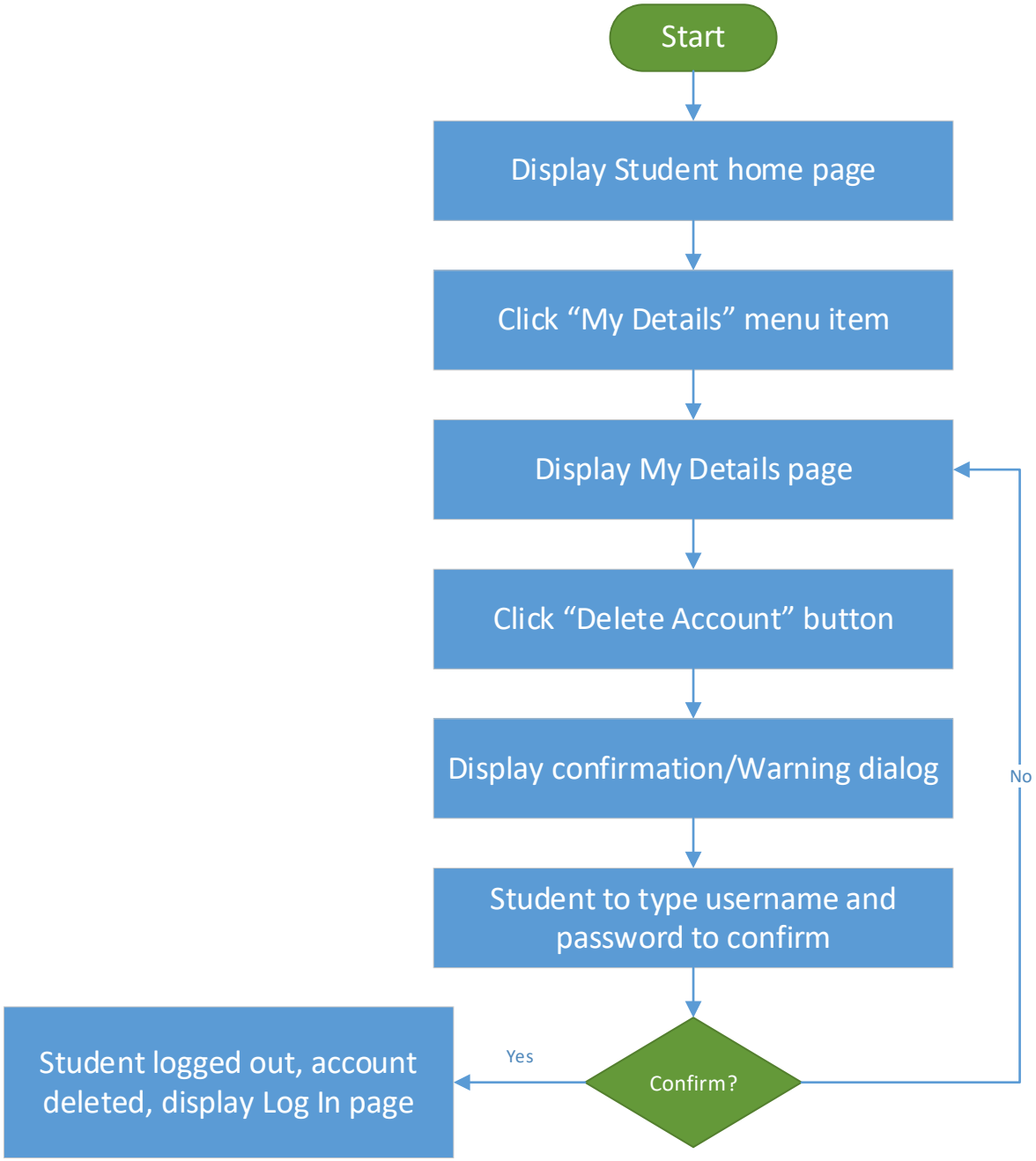
User Flow – Edit Goal (Teacher)



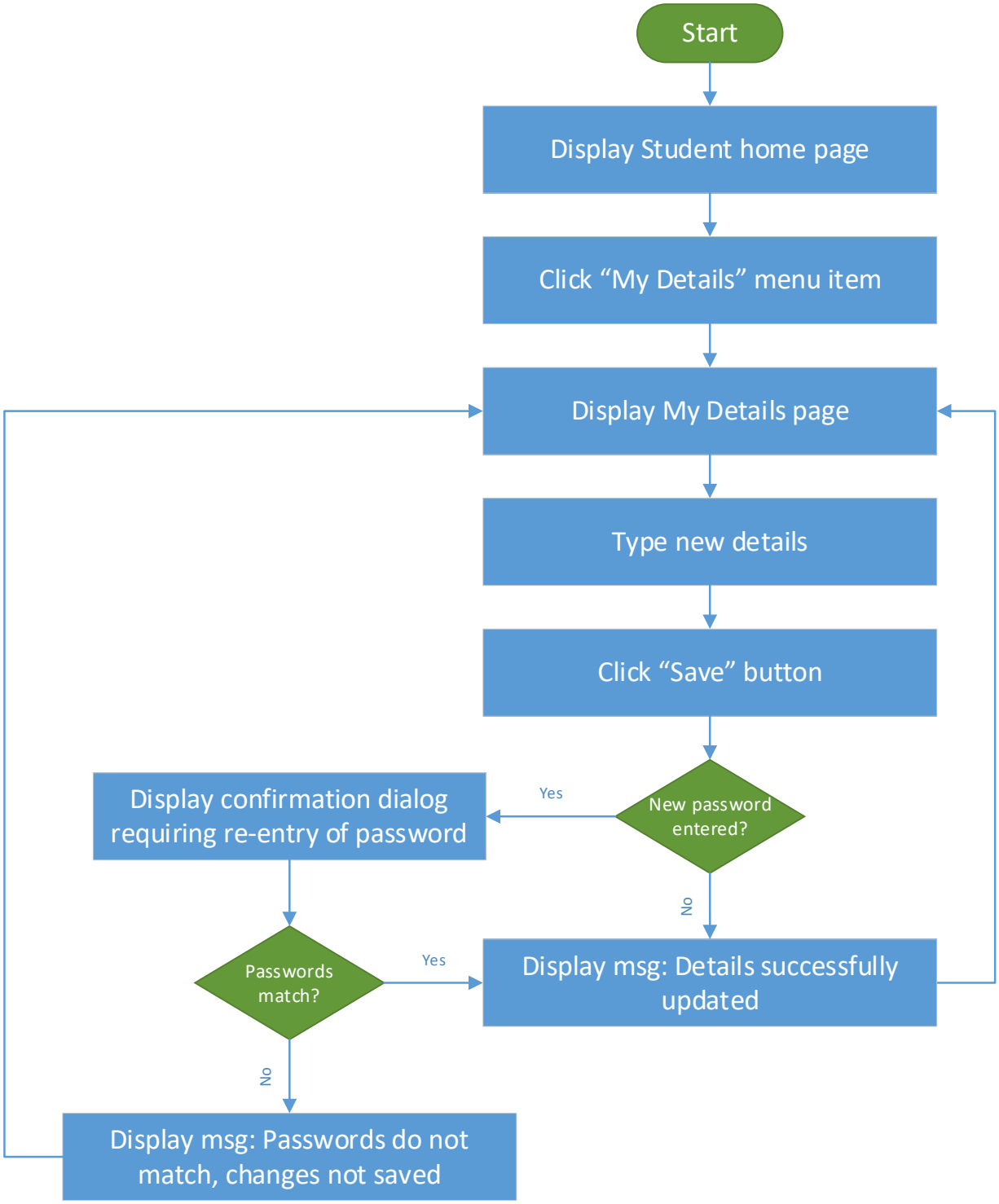
User Flow – Change Goal Completion Status (Teacher)



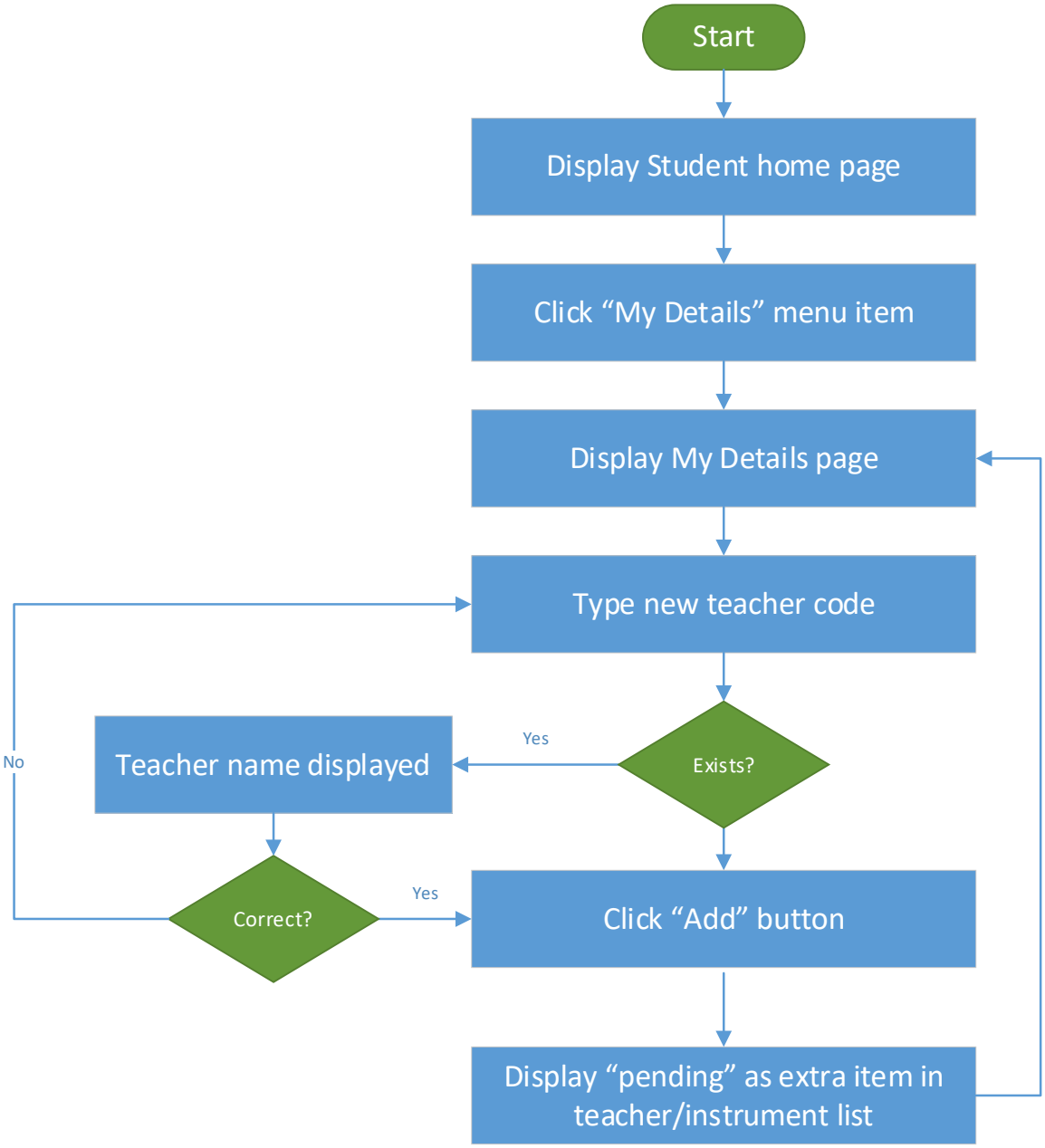
User Flow – Delete Account (Student)



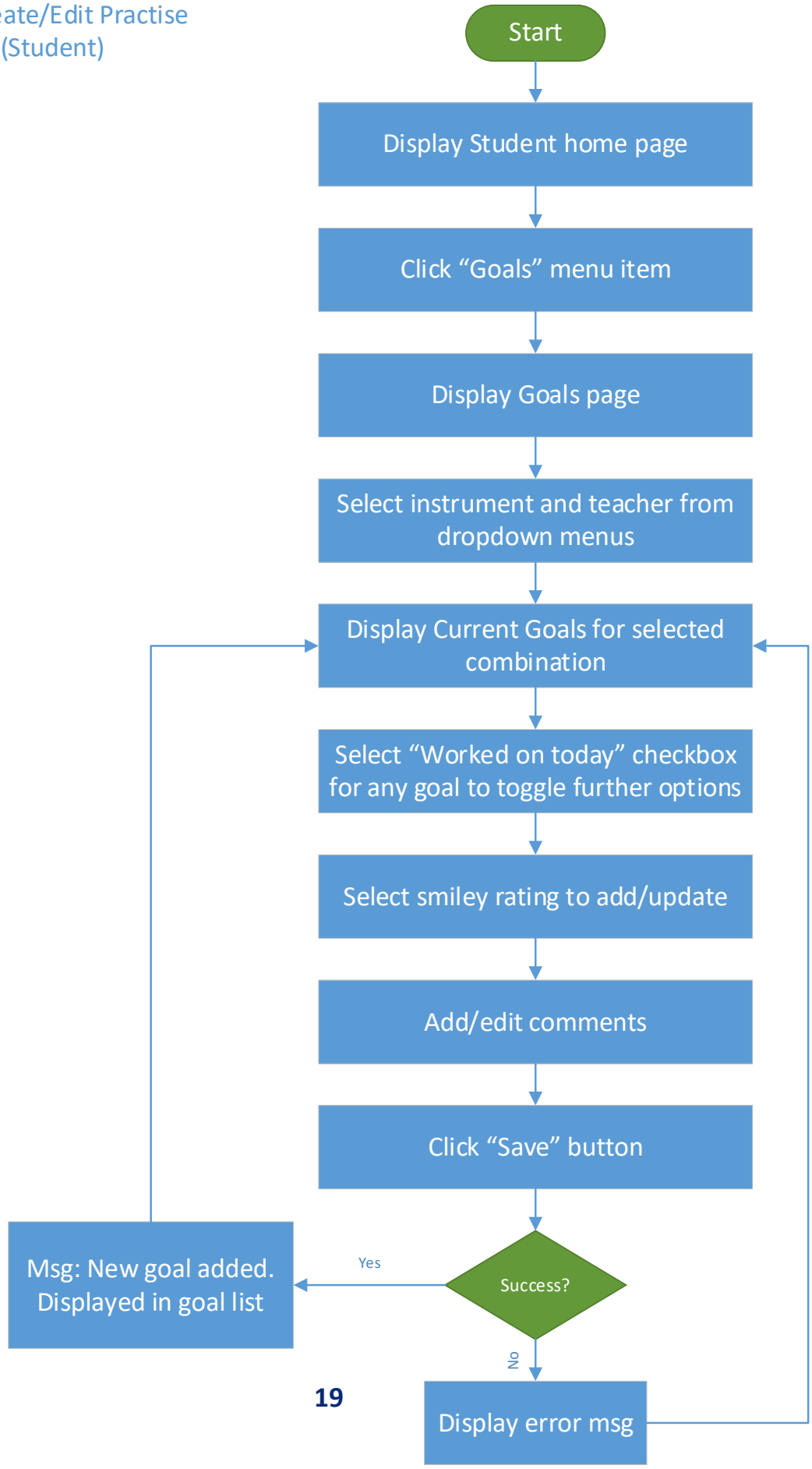
User Flow – Update Account Details (Student)



User Flow – Add Teacher (Student)



User Flow – Create/Edit Practise Session (Student)



User Stories

Title	Goal Assessment	Priority	1
Description	As a teacher, I want to have my students give feedback on their progress, both positive and negative, so that I can address problem areas and praise good effort.		
Acceptance Criteria	Provide students the ability to comment on each goal.		
	Students able to provide self-assessment on a 1-5 scale for each goal.		
	Teacher can read students' comments on their goals.		
		Accepted	

Title	Login/Logout	Priority	1
Description	As an end user, I want to be able to log in and out of my account with a password so that I know my details are secure.		
Acceptance Criteria	Unable to access user information without logging in.		
	Users logged in are only able to access their own data, or in the case of teachers, pertinent records of their own students.		
	Passwords stored securely within the database.		
		Accepted	

Title	Permissions	Priority	1
Description	As a teacher, I want to have the ability to create, read, and update students' goals, where they can only read and comment on them, so that my students are required to discuss them with me each lesson.		
Acceptance Criteria	Teacher login should have CRU ability of student's goals.		
	Student login restricted to reading and updating.		
		Accepted	

Title	Consistent UI	Priority	2
Description	As an end user, I want the website to be easy to navigate and use on any of my devices.		
Acceptance Criteria	Website design is responsive to different screen sizes.		
	All functionality available no matter the screen size or type (e.g., touchscreen or otherwise).		
		Accepted	

Title	Tracking Progress	Priority	2
Description	• As a student, I want to have a visualisation of my achievements up to the present time so that I will have a greater sense of self-efficacy. • As a teacher, I want to see how my students are tracking over time so that I can better tailor my instruction to their needs.		
Acceptance Criteria	Archive completed goals.		
	Numerical and/or graphical representations of achievements.		
	All users able to view list of previous goals.		
		Accepted	

Title	Time Logging	Priority	3
Description	As a teacher, I would like to know how much time students spend practicing each week so as to get an idea of their commitment.		
Acceptance Criteria	Students log day and time spent practicing individual tasks (e.g., scales, repertoire, etc).		
	All users able to see visualisation of cumulative results.		
		Accepted	

Title	Metronome	Priority	4
Description	As a student, I would like a metronome function so that I don't have to purchase a separate device.		
Acceptance Criteria	Replicate all functions of basic metronome.		
	Provide different tones to suit preferences, including whether or not there is a different tone for a specific beat.		
		Accepted	

Scope

In Scope
A client-server web application for music teachers and their students to set goals and track achievements over time, with optional comments and feedback per goal.
Teachers will be able to CRU students' goals, and mark whether they have been completed satisfactorily.
Teachers able to search for students and their goal records.
All users able to see visualisation of achievements to date, and able to search historical data.
Students able to CRUD their personal details/account.
A relational database to store user data for CRUD operations, including encryption for sensitive data.
Hosting everything in the cloud and making it accessible for anyone with internet access.

Out of Scope
Payment and invoicing system.
Metronome function.
Logging time spent practising.
Determining what to do with accounts inactive for a long period of time.
Generating and printing reports.
Teachers able to respond in the app to feedback/comments made by students/parents.
Storing and showing teacher qualifications or details about what grades they teach for which instruments.
Alerting teachers to created or updated comments from students/parents/guardians.

Non-functional Requirements

Security:

- User accounts only accessible via login screen. If a user is not stored in localStorage, then they are automatically redirected to the login screen.
- Passwords not stored as plain text (hashed).
- Access to the database is limited to a specific account (newuser) with only some permissions (SELECT, INSERT, UPDATE, DELETE, EXECUTE, SHOW VIEW. No DDL rights).

Efficiency:

- Software should be easy and intuitive to use, as children will be ideally using it as well as adults.
- The application should respond quickly to user requests, achieved through caching and React which renders components efficiently.
- Though it's not critical in function where perhaps a medical application may be, by hosting the database and web app in the cloud, there's the ability to automate backup instances.
- Well-written and separated modules would help ease maintainability. Writing tests for each function would also assist in this matter.

Project Planning

Capstone Project			Powered by monday.com
Week 1			
Name	**Person**	**Status**	**Date**
Finalise concept	Courtney	Done	2022-09-02
Create architecture diagram	Courtney	Done	2022-09-04
Write user stories	Courtney	Done	2022-09-04
			2022-09-02 to 2022-09-04
Week 2			
Name	**Person**	**Status**	**Date**
Write scope section	Courtney	Done	2022-09-05
Create draft Figma wireframe	Courtney	Done	2022-09-06
Create draft database model	Courtney	Done	2022-09-07
Review and finalise Figma navigation	Courtney	Done	2022-09-07
Review and implement database desig	Courtney	Done	2022-09-08
Finish user flow diagrams	Courtney	Done	2022-09-08
Populate database with test data	Courtney	Done	2022-09-08
Write and test database functions	Courtney	Done	2022-09-10
			2022-09-05 to 2022-09-10
Week 3			
Name	**Person**	**Status**	**Date**
Find and implement MUI template	Courtney	Done	2022-09-13
Customise UI	Courtney	Working on it	2022-09-14
Create GitHub branches, MVC	Courtney	Done	2022-09-16
Sign In	Courtney	Done	2022-09-16
Sign Up	Courtney		2022-09-16
			2022-09-13 to 2022-09-16
Week 4			
Name	**Person**	**Status**	**Date**
Teacher home page	Courtney	Done	2022-09-19
Teacher goals page	Courtney	Working on it	2022-09-20
Student home page	Courtney		2022-09-21
Student details page	Courtney		2022-09-22
Student goals page	Courtney		2022-09-23
Write unit tests	Courtney		2022-09-24
Write README	Courtney	Working on it	2022-09-25
Review and finalise documentation	Courtney	Done	2022-09-25
Review code - tidy and well-commente	Courtney	Working on it	2022-09-25
			2022-09-19 to 2022-09-25

Testing Strategy

The strategy used for testing the functionality of the application began at the database level, with more testing at each level between there and the UI, ensuring connections were correct and functional. See below:

Step One:

- After creating the database, test data was inserted to check basic CRUD operations (see .csv files in MySql_Scripts folder).
- After designing the UI and deciding on required functionality, procedures for these were written and tested using the previously inserted data (see “methods” in MySql_Scripts folder).

Step Two:

- Postman was used to test CRUD operations using the app’s specified routes.

Step Three:

- Performing CRUD operations using the UI.
- Testing within the application code was primarily utilising console.log and inspecting the rendered UI.

Edge cases were given lower priority for the initial release, however coming updates will seek to address these issues. For example, if a user type is logged in and knows the URL for the other user type, they can switch to that UI without logging in as that use type. Not much can be done, but it’s still something that needs to be addressed.

Implementation

Deploying to a cloud hosting service allows for greater flexibility and reliability than on-premise hosting, however server running costs need to be considered either way, and thus the ongoing use of Kicker would require user payment to cover these costs, as well as ongoing maintenance and improvement.

End-to-end Solution

The software has only met some objectives thus far, and more time needs to be invested to fully satisfy the project requirements.

References

Code for both the front and back end, SQL scripts, and .csv files hosted on GitHub.
GitHub link: https://github.com/CourtneyWebber/Kicker

Tools:

- MySQL Workbench 8.0 CE – creating and testing database
- Postman – testing APIs
- Visual Studio Community 2022 – IDE
- Monday (monday.com) – project scheduling
- Microsoft Visio 2016 – diagrams (architecture, user flow, Chen)
- Visual Paradigm 17.0 – database ERD
- Figma – UI design
- MS Paint – logo creation

Frameworks:

- Node.js
- Express.js (for Node.js)

Libraries:

- React.js
- Moment.js
- Material UI (MUI)
- Axios
- bcrypt

Cross-Origin support:

- CORS