Chapter 1

Introduction

Legal driving license acquisition takes a long time. This begins by securing a provisional license, followed by passing both the theory and practical exams. The difficulty mostly stems from the examinations. In the UK, an increasing number of young adults are pursuing full driving licenses each year, reflecting a growing interest in driving.

As of 2024, there are a variety of resources available for new learners such as tools and materials designed to support learning. The Driver and Vehicle Standards Agency (DVSA) published the Official Highway Code Book, which consists of a comprehensive collection of road rules and regulations. Additionally, the DVSA also released an official driving theory learner app, allowing individuals to study at their own pace on their devices. Beyond this, the Internet, YouTube videos and digital libraries are accessible with the relevant material. With such a range of materials available, it is expected that learners can pass their Driving Theory tests easily.

Despite this, research conducted by Scott Le Vine and John Polak indicate that the learning experience for Driving Theory is significantly unsatisfactory. Their study explores why many "young adults delay or forgo acquiring a driving license". Based on the motivations provided by young adults who are not fully licensed, they identified several key factors. The common reasons drawn are that driving is often considered a "low priority", many feel "too busy to learn," or they lack sufficient interest. Additionally, their analysis shows that young adults spend an average of 1.7 years learning to drive after reaching the eligible age of 17. Given that people are preoccupied with other responsibilities, the considerable time needed to learn can be seen as a deterrent to fully commit to it.

According to Figure x, the pass rate for the UK driving theory test (Category B) peaked during the global pandemic in 2020 because of the lenient exam conditions at the time with COVID-19 social distancing policies in place. However, it has been steadily declining as lockdown has slowly been lifted, at a rate of 0.6%. Even with the resources provided, the pass rate in 2024 is currently under 50%. This trend strongly suggests that there may be issues with the Driving Theory learning process. Therefore, it is critical to study the current methodologies in the learning process, especially from the computer science perspective.

1.1 Aims

The research aim for this project is to be able to improve upon the learning experience for the Driving Theory test for **category B licenses only**. To achieve this, the following objectives need to be met beforehand:

- 1. Identify factors why the current learning experience is poor.
- 2. Create an application as a tool for driving theory revision.
- 3. Evaluate how effective this is with a survey.

Once the factors are identified and written into requirements for the development of the application, they can be tested for several non-drivers willing to learn. Therefore, the learning experience is now improved.

1.2 Time Management

In figure x below, this is the Gantt chart of the project that is be the project plan in a timeline from October 2023 to May 2024. It also includes key milestones that need to be achieved in steps and subtasks for this project.

Chapter 2

Literature Review

It is important to identify the factors that affect the learning experience, which is the first research objective of the project. With these factors in mind, it is also worth exploring the current approaches and finding gaps in the literature, which is covered in this chapter. This will contribute to a solution that will help those learning Driving Theory.

2.1 Kolb's Learning Cycle

The foundation to address the problem is to investigate how people learn in general. Thus Learning Cycles, which are frameworks that abstract the phases of the learning process, are a key concept to begin with. This methodical technique allows people to efficiently absorb and remember new information and skills.

Kolb's Learning Cycle is developed by educational theorist David Kolb. It is an experiential-based cycle illustrating how individuals gain deeper understanding of concepts by means of *repetition*. Kolb depicts the cycle as a spiral. When learners revisit a topic or concept, they more deeply and gain more depth, maturity, and understanding each time. This cycle is broad enough that generalises across a wide range of learners. The stages of the cycle are:

1. **Concrete Experience:** Have a new experience or reinterpret an existing one.
2. **Reflective Observation:** Reflect on the experience.
3. **Abstract Conceptualisation:** Form or change theories based on reflections.
4. **Active Experimentation:** Apply the theory to see any changes.

Kolb's Learning Cycle can be applied to learning how to drive. As shown in Figure x above, a learner faces a "yield" road sign, which instructs the driver to give way to incoming vehicles or pedestrians. Firstly, the learner observes a licensed driver when they stop in response to the sign. Next, the learner then reflects on the experience and refer to the sign's definition in Driving Theory material for confirmation. Finally, the learner applies this knowledge in future driving experiences, progressively improving their understanding.

Nonetheless, Kolb's learning cycle has faced criticism in recent papers, such as in John Boyd's work. Boyd argued that the model is too "linear" and "rigid," which could, in turn, slow the learning process. Kolb's framework emphasizes a sequential progression through each stage. This means that if a mistake happens during the "active experimentation" phase, the learner may need to go over another entire cycle to work on the mistake instead of progressing. For instance, in the context of driving, a learner might encounter another "yield" sign in an unfamiliar setting and forget the correct response. This scenario highlights the limitations of Kolb's rigid structure, as the learner may struggle to adapt.

Furthermore, Kolb's learning cycle assumes a continuous learning flow, which doesn't account for time in between cycle stages. This presumes that all learners hold perfect motivation throughout the process. In reality, individuals often need time to mentally recover before they can continue to learn. This makes handling mistakes within the cycle require even more time with mental breaks, which can be discouraging due to the noteworthy time and effort. Reiterating Scott Le Vine et al, it is important to recall many individuals lack either the time or the motivation to dedicate themselves fully to learning how to drive, further highlighting this challenge.

2.2 The Ebbinghaus Forgetting Curve

Another major problem for the learning experience of Driving Theory is that the subject is very content-heavy. Large volumes of material can make learning cycles extremely overwhelming and demotivating. It'll be very difficult to manage the information, to a point where the quality of learning diminishes. Since this is memory-based, this could potentially lead to a rote understanding rather than a true understanding.

This presents a significant challenge for new learners, as they must consistently review Driving Theory on a regular schedule to retain information effectively. According to the forgetting curve, remembering material for the Theory Exam requires not only learning new content but also revisiting and revising it frequently to remember for the Driving Theory exam.

This aligns with the concepts and ideas introduced by psychologist Hermann Ebbinghaus, who described how memory declines over time, a phenomenon known as the "forgetting curve." The theory discloses that a large amount of newly absorbed information is forgotten rapidly, especially within the first few hours. Since one can easily forget what they have learned, this can result in their motivation to plummet.

The curve shape demonstrates how quickly learners forget information initially, but then the forgetting rate flattens over time. The rapid decline shows that memory retention drops exponentially after learning new content. Because of this problem, Ebbinghaus has endorsed the concept of **spaced repetition** that mitigates the forgetting process. Every review slows down the rate which in turn strengthens memory retention. This means the time in between reviews increases due to the information being more encoded in the long-term memory.

Therefore, John Boyd has endorsed a more agile mindset and philosophy to learn as efficiently as possible. It's "the need to be flexible within a given system, and to keep one's orientation matched to the real world". He believes in instant feedback so the previously learned knowledge can be reinforced and contextualised further, rather than statically learning the road rules one by one. In a world where you need to study intensively before an exam date, continuous feedback is very vital.

2.3 Learner applications

There are various approaches to deal with memory retention and motivation with active recall, such as flashcards, mind-maps and anything that involved visual learning. But, one mainstream solution for routine-based learning is the use of learner applications. These are digital tools, educational apps, hands-on exercises, or interactive methods that are made to help learners acquire knowledge and develop skills more effectively.

Currently, numerous driving theory learner applications are available, including the Official DVSA Theory Test Kit, Driving Theory Test 2024 by Driving Test Success, and Driving Theory Test 4 in 1. This segment examines the features of these applications, highlighting key characteristics and identifying common elements that contribute to their effectiveness across a diverse range of users. These Driver-learner apps provide a comprehensive suite of features to equip users with the knowledge and skills to pass their driving theory and practical tests, which include:

- Databases of up-to-date practice questions.
- Mock exams designed to simulate the conditions of the actual test.
- Hazard perception training with interactive video scenarios.
- Road sign quizzes with a comprehensive selection of signs and their meanings.
- Step-by-step tutorials through essential driving manoeuvres.

Additionally, these apps also feature robust progress-tracking tools that monitor performance, highlight weak areas, and adapt learning paths to individual needs. This approach is vital as it addresses Boyd's critiques of Kolb's learning cycle, focusing on the need for dynamic adaptation to a learner's weaknesses. This helps users to identify where to continue with their learning, keeping them engaged and motivated. By emphasizing continuous feedback and iterative decision-making, it supports a more responsive and personalized learning process, ensuring that learners can adjust and improve effectively.

A key characteristic outside of the traditional learning styles all the applications have in common is their reliance on gamification. By incorporating achievements, leaderboards, daily challenges, and progress-based rewards, they transform the learning process for Driving Theory into an interactive, game-like experience. This characteristic is widely used to uplift learner's motivation consistently. Since this is also implementing spaced repetition, this also helps to retain information prevent the learner to forget what they have learned on a regular basis.

2.4 Gamification

In order to address the motivation and memory retention issues for learning how to drive, Gamification in learning integrates game-like elements into the educational experience. Learners can be motivated and engaged to practice consistently daily. This part aims to investigate the case studies of the effectiveness of gamification implemented in learner applications.

Gamification elements "increases the learners' interest" which is crucial for engaging with the Driving Theory learning process. Kačerauskas et al. highlight that gamification boosts motivation, supports self-organization and iterative learning in educational contexts. However, its primary limitation lies in its "temporality". As users become more experienced with the gamified service, its effectiveness diminishes, potentially reducing long-term engagement.

Similarly, a systematic review by M. Shortt et al. on Duolingo found a generally positive correlation between gamification and learning outcomes but noted confounding variables. For example, mobile devices, while convenient, introduce distractions such as access to social media, negatively impacting focus. Additionally, repetitive content within gamified systems led some learners to experience "diminished motivation". These findings emphasize the challenges of maintaining sustained engagement and focus in mobile-based learning environments.

Therefore, one major issue with the design of many current learner apps is temporality and lack of replayability. As a result, learners may become bored to restudy, which ultimately contributes to the forgetting the material, lessening the overall learning experience.

2.5 Replayability

As discussed, many learner apps struggle with temporality, where repeated use leads to diminishing engagement and ultimately less effective learning. This section explores the concept of replayability and how it can be integrated into learner apps to ensure prolonged use of gamified learning tools.

The research study by Rose Adetunji, explores how replayability impacts educational game design, focusing on how to allow students to consistently deepen understanding through repeated reinforcement. With the recent rise of articles on gamification, there is a gap in the literature on the impact of replayability. The study suggests the importance of ”supporting repeated play and diverse learning experiences.” She has found that the key factors that drive a player to replay a game are:

1. **Variety/Randomness:** Adding variation to scenarios for deepened understanding.
2. **Goals/Completion:** Providing meaningful feedback keeps learners motivated.
3. **Difficulty:** Should present new challenges with each replay to increase in difficulty.
4. **Social aspects:** Playing alongside other players for influence to engage in the game.

Replayability allows learners to interact with material in various ways, reinforcing memory and understanding through practice in increasingly complex scenarios. An experiment tested a replayable RPG, incorporating elements similar to Adetunji's study. Player Motivations varied as some wanted to explore game mechanics (Variety/Randomness), others aimed to master the game (Goals/Completion), and some were driven by the desire to overcome challenges (Difficulty).

By augmenting more replayability with the factors discussed, the game-design in the current Driving Theory Learner applications can be improved upon which can mitigate the loss of motivation. The whole point of the application being as replayable as possible is to help the learner to carry out spaced repetition to remember as much as possible for their Driving Theory exams.

2.6 Framing the problem in Machine Learning

Now that the factors have been identified, it's now possible to refine learner applications to help struggling learners. To further enhance the learning experience, machine learning can be applied as they can provide adaptive, personalized learning experiences.

It is required to use a learner's performance over a study session in order to evaluate and tailor the appropriate feedback. Predictions also need to be made in order to recommend topics/questions based on which are their general strengths and weaknesses. Therefore this is a **Supervised learning** task that utilises **Multivariate Regression.**

Supervised learning is a machine learning technique that uses labelled input output datasets to train algorithms to recognize patterns and predict outcomes. For this case, the inputs/features would be the different questions as well as other performance metrics such as answer accuracy, and time spent. Whereas a typical target variable will be the the scoring on the question. In fact, 88% of found studies that apply machine learning in an educational setting, have applied supervised learning, which endorses the use of the model even more.

Regression is a type of Supervised learning technique that predicts continuous values, in this instance that would be the performance scores for future mock Driving Theory exams. This problem is considered **Multivariate** as the evaluation is based across multiple Driving Theory topics, time spent on questions and scores on quizzes. It is important to account to make the model flexible for a range of learners. This is essential in order to evaluate what topics and questions need more practice.

Several regression algorithms, such as Linear Regression, Logistic Regression, and Random Forest, offer robust methods for analyzing data. However, given the extensive dataset of questions and topics involved, combined with the unpredictability in learners' progression, Random Forest proves particularly effective. It excels at capturing complex relationships within the data and provides a more comprehensive evaluation of learner performance, making it well-suited for this context.

2.7 Applying the Random Forest Algorithm

Random Forest has be used in various statistical contexts such as Prediction of default Credit Card activity, medicine and economics. This last portion of the review looks into the Random Forest algorithm and how it's potentially applied as a component to this new learner application for Driving Theory.

The building blocks of this algorithm are the concept of **Decision Trees.** They are a flowchart-like structure, where each internal node represents a decision based on a feature, each branch represents the outcome of the decision and the leaf node represents the final decision.

Therefore, Random Forest Algorithm combines the predictions of multiple decision trees during training and combine their predictions to produce an accurate and robust model. How it works:

1. **Bootstrap Sampling:** create subsets of training data by sampling with replacement.
2. **Random Feature Selection:** In each split in a decision tree, use random features.
3. **Ensemble Prediction:** Prediction based on the decision trees' final decision.

In order to increase the precision, one model can be utilised per topic. This means there will be predictions based on the performance on all the questions under the same category, so that the model will avoid too much generalisation. An example usage in personalised application to predict a learner's score in a category can use the following features like time spent, accuracy and difficulty level of a question. This is due to the differing nature of the Driving Theory Topics. For example, Hazard Perception heavily depends on the accurate timing spotting the developing hazard whereas a multiple choice question needs to be answered quickly and precisely.

2.8 Summary

This chapter showcases a problem in the learning experience for Driving Theory. The main factors are that people need motivation and are prone to forgetting material they've studied, especially for how immense Driving Theory is as a subject. There are flaws in general learning models like Kolb's learning cycle. Boyd's critique underscores the cycle's inflexibility and rigid structure, emphasizing the need to balance mental rest and consistent information reinforcement. These shortcomings contribute to challenges like the forgetting curve, where learners struggle with retention over time, leading to diminished motivation.

Current solutions, such as Driving Theory learner apps, heavily utilize gamification to improve memory retention and engagement. While effective initially, limited replayability often results in boredom, reduced motivation, and decreased learning outcomes. To combat these issues, Supervised Learning with Multivariate Regression through **Random Forest** can be applied, enabling deeper monitoring of learner performance, personalized feedback, and sustained engagement.

With the background research addressing all the current problems of the learning experience of Driving Theory, this can help establish the design choices for the newly refined Driving Theory learner application. It will utilise Gamification and attempt to improve the replayabilty and give continuous, precise and constructive feedback. These are especially important requirements to include for this application in order to keep learners motivated and avoid forgetting. Therefore, the learning experience can be improved.

Chapter 3

Application

Now with the literature review, the requirements and user stories can be generated in order to improve the learning experience. This chapter goes in depth of the design choices of this newly refined application.

3.1 Requirements

(To be refined in final report)

ID	**Requirement**	**Reference**
#1	New users must be able register and login into the system with a username and password.	See section 2.3
#2	Application must provide a structured training session: (a) A warm up phrase with easy questions. (b) Practice round with hazard perception. (c) Practice round with road signs. (d) Practice round with completely random questions. (e) After the session provide performance feedback. (f) Scores are calculated based on accuracy and time taken.	See section 2.2
#3	Users must have a profile that displays their game statistics: (a) Current level. (b) Their daily streak. (c) Achievements. (d) Average reaction time for hazard perception.	See section 2.4

ID	**Requirement**	**Reference**
#4	Allow users to add each other as friends through a friend code.	See section 2.5
#5	Users must have a clear overview of their learning progress: (a) Highlight the mastery of each topic. (b) History of *recent* performances from previous training sessions.	See section 2.2
#6	Each question must have multiple variations of it's wordings, along with rephrased versions of its (in)correct answer(s).	See section 2.5
#7	Each question must be labelled with a distinct topic.	See section 2.3
#8	The application must have a database system accessed via an API. User and question information are stored in the database separately.	See section 2.3
#9	Application must allow users to practice on specific topics.	See section 2.1
#10	Application must indicate any actions taken place e.g. Friend request, Progress saved.	See section 2.3
#11	In the first training sessions, the application must recommend the user random topics and questions.	See section 2.1
#12	In future training sessions, the application must recommend the user more of their weakest topics and questions, but still keep the randomness.	See section 2.6
#13	The user must be able to access the road signs and their meanings from the database via the API function.	See section 2.1
#14	The application must hold a weekly leaderboard, based on earned scores.	See section 2.4
#15	System must support the hazard perception test videos, they can click on the video etc.	See section 2.3
#16	System must support the hazard perception test videos.	See section 2.3
#17	Users can input the date of their Theory Exam for the app to recommend how often to practice.	See section 2.3
#18	The application must be able to give mock exams based on the exam structure of the actual Driving Theory exam.	See section 2.3

Some additional requirements, which could be implemented if some time is left over:

ID	**Optional Requirement**	**Reference**
#19	Application could have a friendly 1 vs 1 quiz battle game: (a) Rapid round of completely random questions of a set difficulty. (b) The player with the fastest and most accurate answers win. (c) Game performance displayed in the end.	See section 2.4

3.2 Design

UML DIAGRAMS (YET TO BE DEVELOPED):

- Class Diagram of application.
- Activity Diagrams of the different simulations of the application.

TOOLS (TO BE FINALISED IN FINAL REPORT):

- Everything will be coded in Visual Studio Code, it's extensions support the following tools.
- *Microsoft Azure service* will be used for the back-end API and database:
  - API functions will be coded in python.
  - Microsoft CosmosDB will be used for the database.
- *Javascript* will be used for the front-end and server-client interactions:
  - Views are coded in *VueJS*
  - *VueJS DevTools* helps with developing and debugging.
  - API calls are done here.
- Machine Learning model coded in *python* with *Data.Frame*:
  - API accesses this model and sends response to server.

3.3 Implementation and Testing

Implementation will be carried out with the Agile development approach. There various components that need to be implemented in a certain order as some components are dependant on another and vice versa. The whole development will be test-driven. The iterations are as follows:

PROTOTYPES:

- The Azure API / Client-Server component prototypes: previous similar projects.
- Machine learning: Experimenting with Data.Frame imports in python.

ITERATION 1: BACK-END

- The Azure API functions.
- The machine learning model.
- The CosmosDB database structure.
- **TESTING:**
  - Python Unit tests are done with JUnit.
  - Each API function will have a test.
  - Proxy Objects can be used to manipulate the database for testing.
  - Machine Learning model will be given mock user data to work with.

ITERATION 2: FRONT-END

- Server-Client interactions.
- Server-API interactions.
- Client's UI.
- **TESTING:**
  - These are tested once back-end is complete.
  - Changing the states of the client and server to check the appropriate changes are made.

Chapter 4

Evaluation

This section will be how the application turned out and to see if the learning experience for Driving Theory has improved. This will be done with **Quantitative Research Methods**. A Questionnaire that is tailored towards people who currently have a *Provisional license* as this is the scope of the application. The Questionnaire is as follows:

- User Experience:
  - How easy was it to navigate through the application?
  - Were you able to find the features or sections you needed quickly? Explain why.
  - Did the app feel intuitive to use? Why or why not?
- Content Quality:
  - Did you find the questions in the app relevant and up-to-date with current driving theory?
  - Were the explanations for correct answers helpful and clear?
  - Was there any content you felt was missing or could be improved?
- Learning Effectiveness:
  - How effective was the app in helping you understand driving theory concepts?
  - Did the app help you identify areas where you needed improvement?
  - After using the app, how confident do you feel about passing a driving theory test?
- Features and Functionality:
  - Did you encounter any technical issues (e.g., crashes, slow loading times)?
  - How would you rate the performance of the practice tests and simulations?
- General Feedback:
  - What was your favourite feature of the app? Why?
  - What was your least favourite feature or aspect of the app? Why?
  - Do you have any additional comments, suggestions, or improvements?

4.1 Reflection

(To be written once implemented)

Chapter 5

Conclusions and Future Work

(Any conclusions yet to be written)