OCTOBER 22, 2024

duolingo

Group 8

PRESENTED BY

CLARANNE FECHTER, ELIZA BAIER,
IAN JOHANSON, TINGTING FENG

UNIVERSITY OF UTAH
IS 6420: DATABASE THEORY AND DESIGN
PROFESSOR PUGSLEY

TABLE OF CONTENTS

EXECUTIVE SUMMARY

In today's connected world, it has become increasingly important to communicate with people all over the world. The more languages a person knows, the more connected they are able to be and the more employable they are. The Duolingo app has transformed language learning for people around the world who have access to a smartphone.

Duolingo's goal is to "develop the best education in the world and make it universally available." (Duolingo, n.d.) Since launching in 2012, the app has expanded to include courses in music and math, as well as 40 languages, ranging from commonly spoken languages such as Spanish and Mandarin to less common languages like Welsh and Navajo. (Matt, 2024)

The majority of the application centers on features that support language learning so we focused on those core components as well as a few more auxiliary elements that Duolingo uses to encourage people to continue learning such as notifications, friends, and score statistics. These features have made Duolingo famous as an effective language learning app that provides accessible and tailored education to people around the world.

To capture the core elements of Duolingo's application, we created 13 tables representing existing features and 7 tables representing features we recommend that Duolingo implement in the future. The 13 core tables include: Duo User, Friendship, Friend, Enrollment, Course, Lesson, Content Type, Notification, Score, Join League, League, User Plan, and Subscription Plan. The future endeavor tables include: Test, Test Question Information, Test Question, Answer, Product Placement Inclusion, Product Placement, and Client Company.

We then populated these tables with model data to show how a user might interact with the features of the application. This example data, while limited compared to Duolingo's 100 million monthly active users, shows the interactive features of the application and the learning process through which users transform their language abilities. (Duolingo, 2024)

DUOLINGO GENERAL DESCRIPTION

Background and History

Duolingo is a learning service that helps people learn languages, math, and music. The company was founded in 2011 by Luis von Ahn and Severin Hacker. Their app was released in 2012 and has become one of the most popular ways to learn a language. Duolingo's headquarters are in Pittsburg, Pennsylvania. The company has their main Duolingo app, Duolingo ABC, which is for kids to learn how to read, and their website. They have about 500 million users worldwide and about 100 million monthly active users (Duolingo, n.d.).

Vision and Objectives

Duolingo's vision is to develop the best education in the world and make it universally available. Their global team works together to make language learning fun, free, and effective for anyone who wants to learn, wherever they are (Duolingo, n.d.). Helping people learn in a way that is tailored for them and in a way to keep them motivated by making the learning experience fun and exciting while allowing the best education to be free for everyone is Duolingo's biggest goal.

Products and Services

Duolingo has three main products, the Duolingo app, Duolingo ABC, and their website which has the Duolingo English Test. With the Duolingo app, customers are able to learn over 40 different languages, while also offering math and music courses where customers can learn at their own pace. Duolingo ABC is an app intended to help children learn how to read. On their website, customers are also able to have a fun learning experience just like on the app. However, on the website Duolingo offers an English Test where customers can certify their English proficiency. With the English test, customers are able to practice for free, but have to purchase the test for 65 dollars. For learning on the app or website, they offer 3 subscription plans for users: Free, Super, and Super Family. With Super and Super Family, customers get unlimited access to lessons, more listening and speaking practice and no advertisements. Customers also have the option to buy an add on called Max where they can speak with real world speakers.

DUOLINGO GENERAL DESCRIPTION

Duolingo Transactional Databases

Duolingo uses transactional databases to create new lessons and courses for users, put users in leagues, track subscription plans and in-app purchases, user profile information and updates, notifications, and keep track of user scores. For our database, we decided to focus on user profile information since Duolingo is meant to be customized for the user. We then added a focus in the courses and lessons since the app customizes the learning experience for the customer. We structured our database with tables that establish the connections between the user and the app features, supporting customization and personalized learning paths.

PRIORITIZED REQUIREMENTS

To establish the scope of our project, we've identified the most critical features that are essential for an outstanding user experience. The following functionalities are our top priorities:

- Enable users to register and customize their profiles.
- Provide a wide range of courses for users to learn.
- Organize lessons into different levels to promote step-by-step learning.
- Allow users to track their learning progress, scores, and milestones.
- Include a feature for users to connect with friends and compete in leagues to enhance engagement.
- Offer subscription plans.

After a thorough analysis of the Duolingo app, we've determined that the following specific database tables are necessary to support these features:

- duo_user
- user_plan
- course
- lesson
- score
- subscription_plan
- join_league
- league
- friend
- friendship
- notification
- enrollment
- test
- test_question
- test_question_info
- test_answer
- product_placement
- product_placement_inclusion
- content_type
- client_company

By implementing these tables, we can facilitate account creation with essential details such as user information, subscription plans, and lessons. Furthermore, we'll manage databases for courses, lessons, user progress tracking, scores, friendships, and notifications, covering all fundamental aspects of a language learning platform.

In this project, we've developed these tables within the 'duolingo_db' database and successfully populated them using SQL. Our current database includes 10 users, 10 courses, 10 lessons, over 100 test questions and answers, and a wide array of related data to support user engagement and learning.

The project has substantial potential for future enhancements, and we plan to introduce additional features moving forward.

NEW VENTURES

New Venture

We have introduced two exciting new features to our phone app that utilize our existing database infrastructure.

Language Certification Tests:

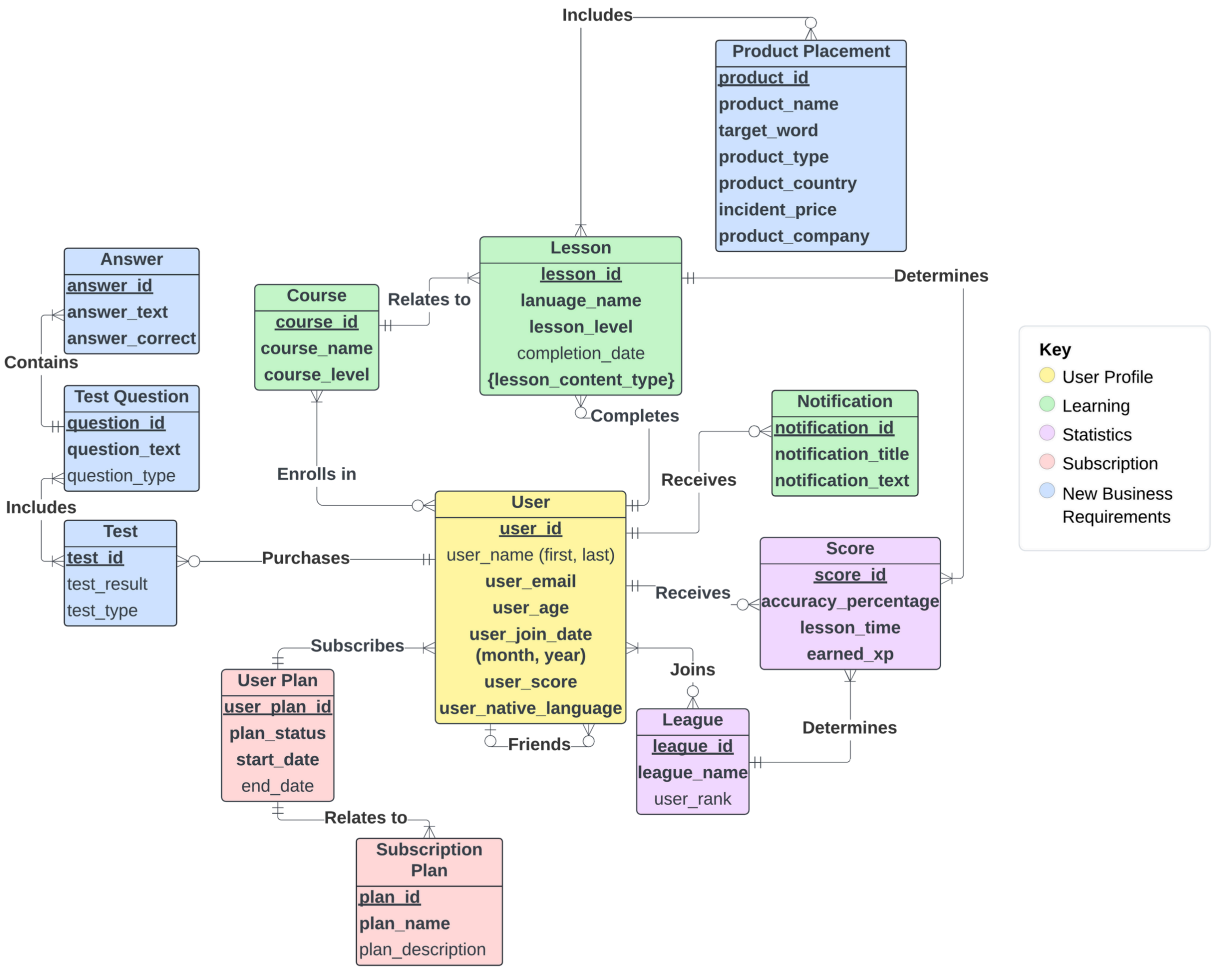
Language certification tests for the Duolingo English Test (DET) let users practice for the official exam through a dedicated section in the app that replicates the exam's format, question types, and time limits. This provides a realistic practice environment where users can test their knowledge, receive instant feedback, and assess their readiness for the actual exam. The feature offers flexibility, enabling users to take certification test preparation anytime, anywhere, even without a computer. It also reduces exam anxiety by familiarizing users with the exam structure in a low-pressure setting.
In the future, this feature could expand to include preparation for other language certification exams, such as TOPIK for Korean or JLPT for Japanese. This would position Duolingo as a leading platform for language exam preparation. Additionally, Duolingo could introduce premium certification test packages with advanced features, creating new revenue streams while continuing to offer core free services.

Product Placement Feature:

We have implemented a product placement feature that allows companies to integrate their branded words into appropriate lessons. This helps users become more familiar with these words, similar to how 'Kleenex' has become synonymous with tissues, thereby promoting brand recognition while benefiting users by exposing them to real-world vocabulary. This feature creates a unique opportunity for companies to engage directly with learners in an educational setting, enhancing brand recall in a meaningful and organic way. By seamlessly embedding branded terms into lessons, companies can foster a stronger connection with potential customers, while learners benefit from exposure to practical, everyday language. This approach not only enriches the learning experience but also bridges the gap between education and real-world application, making the lessons more relatable and memorable. Additionally, the product placement feature introduces a potential revenue stream, as companies can pay to have their brands featured, thus contributing to the financial sustainability of the platform.

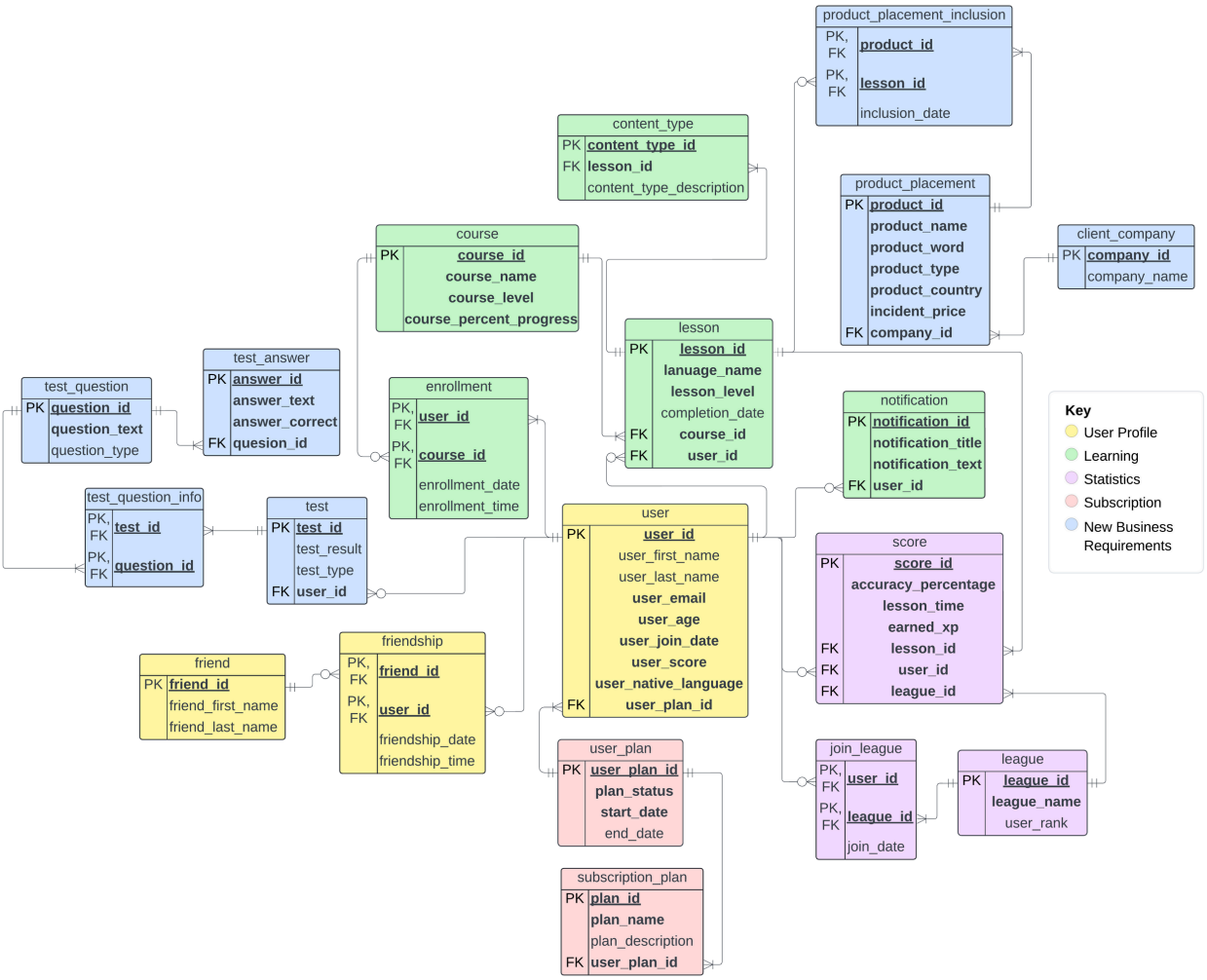
CONCEPTUAL MODEL

To develop the conceptual model, we made several revisions, including some at Professor Pugsley's suggestion. The 'User' entity is the center of the conceptual model because of the tailored nature of the learning model. Each decision the user makes affects all of the other entities in the diagram. The model focuses on users, learning, statistics, subscriptions, and two new business requirements. Our two new business requirements are shown in blue and are 1) product placement and 2) transitioning the English test that is currently available on the website to availability on the app. This particular venture could continue to expand as Duolingo works with TOEFL and other international language test organizations to offer certified tests in English and other languages.

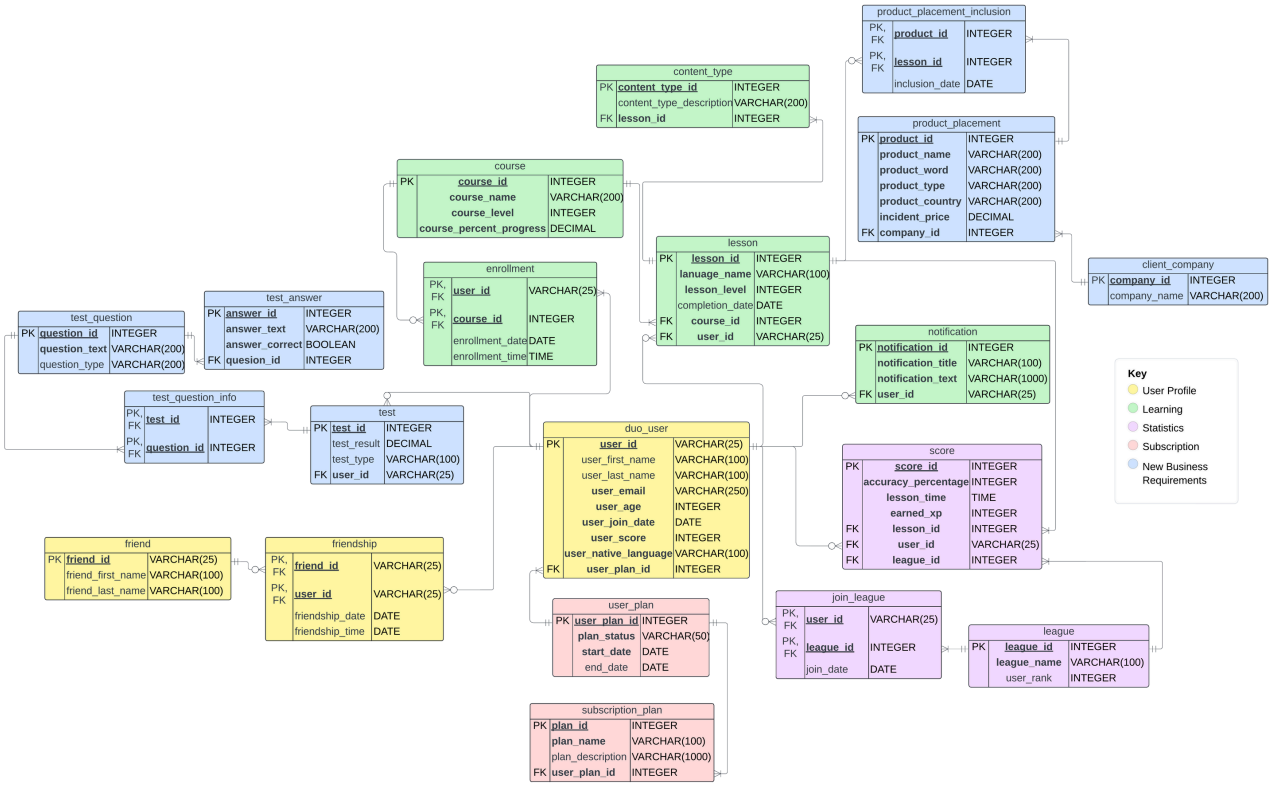


RELATIONAL/LOGICAL MODEL

The next step in developing our database is converting from the conceptual model to the relational model. The following diagram shows the relational model, where primary and foreign keys are assigned to each table, making the relationships between entities more explicit. For each many-to-many relationship, we created bridge tables using composite keys from the related entities. For example, the table test_question_info manages the many-to-many relationship between test and test_question. We also introduced the friendship table to manage self-referential relationships, such as tracking friendships between users. Additionally, new business features, such as Tests and Product Placement, have been integrated into the model. Specific tables like test_question, test_answer, and product_placement_inclusion ensure that these new functionalities are fully captured in the database.



PHYSICAL MODEL



To develop our solution and create a working database, we first examined the Duolingo app to understand the potential business requirements. Then, we created ideas for two new proposed additions to the mobile app and moved on to limit our scope, excluding features that were not central to the function of the app and the learning of users in order to stay within the requirements of the projects. We first limited ourselves to just the mobile app and excluded several other features such as the user avatar, feedback, user achievements, collections, quests, conversations, and the user's feed. These features were secondary to the important learning-focused features we chose. Throughout this process of elimination and afterwards, we completed several drafts of our conceptual model. We also incorporated feedback from the professor on our conceptual and relational model. Our final conceptual model had 12 total entities. Eight of those entities were for the current Duolingo app including: User, Course, Lesson, Notification, Score, League, User Plan, and Subscription Plan. The remaining four entities corresponded to two new features we propose Duolingo add to the mobile app - product placement and the English test that currently exists on Duolingo's website. These four new entities included: Product Placement, Test, Test Question, and Answer.

PHYSICAL MODEL

After finalizing our conceptual model, we created a relational or logical model. We added bridges in place of the many-to-many relationships and designated the appropriate primary and foreign keys in each table.

Finally, to create our physical model, we used our relational logical model as a base. We added in the variable characteristics using the INTEGER data type for all of the id numbers except for the Duo User identification number. We also frequently used the VARCHAR, DATE, and TIME data types to create data for the remaining variables. We used one BOOLEAN type to code our answer choices as correct or not.

After creating the physical database design, we populated the tables with sample data. We did this using ChatGPT to quickly generate data with connected primary and foreign keys. Our final populated tables include 10 users, some with friends and friendship information; 10 sample courses; 10 lessons complete with enrollment information; all 10 Duolingo leagues complete with users in the first 3; 10 score statistics for various users and lessons; 5 sample notifications; 10 client companies and their product words in various languages along with product placement inclusion information; 50 sample test questions with attached information and 94 potential answers; and 4 potential content types for lessons as seen in the Duolingo app.

REQUIREMENTS REVIEW

By the end of our project, we were able to execute almost all of our top priorities and make them functional for our database.

- Users can register their profiles
- Many courses are available for users to learn
- Lessons have different levels
- Users can see their scores and their process though the course
- Users can connect with friends and compete in leagues
- Subscription plans are offered

We decided to leave out certain individual user features that, while personal, were not critical to the app's overall functionality. This allowed us to focus on the core features necessary for the app to run efficiently. We ran into some complexity issues that would not help the functionality of the app and decided that the user avatar, feedback, user achievements, collections, quests, conversations, and the user's feed tables were not needed as these features are out of the scope of our project. We were able to implement every other requirement for our database.

To apply these features into our database, we chose these tables to implement:

- Duo_user - 10 unique rows
- User_plan - 10 unique rows
- Course - 10 unique rows
- Lesson - 10 unique rows
- Score - 10 unique rows
- Subscription_plan - 3 unique rows
- Join_league - 9 unique rows
- League - 10 unique rows
- Friend - 5 unique rows
- Friendship - 5 unique rows
- Notification - 5 unique rows
- Enrollment - 10 unique rows
- Test - 4 unique rows
- Test_question - 50 unique rows
- Test_question_info - 20 unique rows
- Test_answer - 94 unique rows
- Product_placement - 10 unique rows
- Product_placement_inclusion - 10 unique rows
- Content_type - 4 unique rows
- Client_company - 3 unique rows

We completed all of these tables listed above.

ETHICAL CONSIDERATIONS

When dealing with the collection, analysis, and distribution of data in this project, several ethical considerations must be addressed to ensure responsible and transparent handling of user information:

- Data Collection: The collection of user data, such as personal details, learning progress, and subscription information, should be minimized to only what is necessary for providing the intended features. Users must be informed about the types of data being collected and the purpose behind it. Ensuring informed consent is essential for ethical data collection, and users should have the option to opt out of certain data collection practices if possible.
- Data Analysis: Analyzing user data for features like progress tracking, friend interactions, and certification tests can offer valuable insights, but it is crucial to avoid misuse. The data must not be used to manipulate user behavior in exploitative ways, and any analysis should prioritize user benefits, such as improving learning outcomes or enhancing user experience.
- Data Distribution: Sharing user data with third parties, such as companies involved in the product placement feature, raises concerns about privacy and user trust. It is important to ensure that user data is not shared without explicit consent and that any data shared is anonymized to protect individual identities. Transparency regarding data-sharing practices is key to maintaining user trust.
- Security and Oversight: To responsibly manage the dataset, robust security measures are required. This includes encryption of sensitive user data, secure authentication methods, and regular security audits to identify and address vulnerabilities. Oversight mechanisms, such as role-based access control, should be implemented to limit database access to authorized personnel only, reducing the risk of data breaches. Furthermore, regular compliance checks and adherence to data protection regulations (such as GDPR - General Data protection Regulation) are necessary to ensure that data handling practices meet legal and ethical standards.

CONCLUSION

Duolingo is a leading language learning platform, known for its interactive and engaging features. In this project, we enhanced the platform by introducing two new features: Language Certification Tests and Product Placement. These additions are designed to improve both the user experience and Duolingo's business model, expanding its reach while maintaining its educational focus.

Future Features Advantages

Language Certification Tests

Convenience: Take certification tests anytime, anywhere, with no computer needed which is perfect for mobile users.
Realistic Practice: Replicate the format and difficulty of the real exam, helping users familiarize themselves with the exam pace and reducing anxiety.
Instant Feedback: Get immediate feedback to understand strengths and areas for improvement, enabling focused adjustments.
Expansion to Other Languages: Expand certification tests to other languages such as Korean (TOPIK), Japanese (JLPT), or Chinese (HSK), offering users more certification options.
Potential Paid Options: Introduce premium test packages with detailed performance reports and personalized difficulty adjustments, enhancing user experience and driving new revenue streams.

Product Placement

Immersive Learning: Integrate brands and products into the learning process, allowing users to apply language in real-life scenarios, which enhances their learning experience.
Marketing Opportunities: Enable brands to reach a broad audience through Duolingo, providing targeted marketing that seamlessly aligns with language learning content.

Next Steps and Future Endeavors

Next Steps

The obvious next steps for this project would be to model and include the tables we were not able to include because of time constraints and the scope of the project. These tables might support features such as: a user avatar, user feedback, user achievements, collections, quests, conversations, and create a user feed. Additionally, as Duolingo pilots our new venture ideas, there may be additional features that could be added to support these ventures, such as another level of subscription plan for official tests.

CONCLUSION

Future Endeavors

In the future, to fully take advantage of the available market opportunities, Duolingo needs to expand its technical infrastructure to support additional language tests. Duolingo can use market research and data analysis to assess user needs for authentication in different languages. Focus resources on the most marketable language certifications, such as TOPIK in Korean or JLPT in Japanese. In addition, through algorithms and data analysis technology, product placement can be adjusted in real time according to each user's learning behavior and personal preferences. This individualized approach can increase user immersion, while also providing brands with more targeted marketing opportunities.

REFERENCES

- Duolingo. (n.d.). Duolingo English Test. Retrieved from https://englishtest.duolingo.com
- Duolingo. (n.d.). Duolingo Press Room. Retrieved from https://press.duolingo.com/#about.
- Duolingo. (n.d.). About Us. Retrieved from https://www.duolingo.com/approach.
- Duolingo. (2024, August 7). *Duolingo Hits 100M MAUs, Reports 59% DAU growth and 41% Revenue Growth in Second Quarter 2024 - Duolingo, Inc.* Duolingo Investor Relations. Retrieved October 17, 2024, from https://investors.duolingo.com/news-releases/news-release-details/duolingo-hits-100m-maus-reports-59-dau-growth-and-41-revenue
- Matt. (2024, June 27). *The Complete List of EVERY Duolingo Language in 2024*. duoplanet. Retrieved October 17, 2024, from https://duoplanet.com/duolingo-languages-list/

duolingo

OCTOBER 22, 2024

duolingo

Group 8

APPENDIX

UNIVERSITY OF UTAH
IS 6420: DATABASE THEORY AND DESIGN
PROFESSOR PUGSLEY

TIME LOG

Team Member	Total Hours	Description of Work
Claranne Fechter	16.5	• Conceptual, Relational, and Physical models • Report • Presentation • Meeting preparation • Practice presentation • Meetings
Eliza Baier	25.2	• Conceptual, relational, and physical model (+ applying feedback) • Presentation • Meeting preparation • Database and data creation and testing • Report • Meetings
Ian Johanson	19.9	• Business requirement gathering • Conceptual and relational research and model (+ applying feedback) • Database and data creation and testing • Report • Presentation • Meetings
Tingting Feng	16.1	• New business venture ideas • Conceptual, relational, and physical models (+ applying feedback) • Report • Presentation • Meetings

DETAILED TIME LOG

Name	Date	Time Spent (hrs)	Description of Work
Claranne Fechter	8/25/2024	1	Conceptual Model
Tingting Feng	8/25/2024	1	Conceptual Model
Eliza Baier	8/26/2024	1	Conceptual Model
Ian Johanson	8/26/2024	1	Business requirements gathering
ALL	8/29/2024	1	Conceptual Model
Eliza Baier	9/2/2024	1	Conceptual Model
Tingting Feng	9/2/2024	1	New Business Venture Ideas
ALL	9/3/2024	1	Conceptual Model, Entity Expansion Ideas
Eliza Baier	9/4/2024	2.5	Conceptual Model, Presentation
Claranne Fechter	9/10/2024	1	Conceptual Model
ALL	9/17/2024	1	Conceptual Model
Claranne Fechter	9/22/2024	1	Relational/ Physical Model
Tingting Feng	9/22/2024	1	Relational/ Physical Model
ALL	9/23/2024	1	Relational/ Physical Model
Ian Johanson	9/23/2024	1	Conceptual/ Relational research
Eliza Baier	10/4/2024	2.1	Conceptual/Relational Model (applying feedback), Presentation
Tingting Feng	10/4/2024	1.3	Conceptual/Relational Model (applying feedback)
Ian Johanson	10/4/2024	1.1	Conceptual/Relational Model (applying feedback)
Claranne Fechter	10/7/2024	1	Models
ALL	10/8/2024	1	Conceputal model and physical model revision
Ian Johanson	10/8/2024	1	Database Creation Testing
Eliza Baier	10/10/2024	2	Presentation, Meeting Preparation, Database Creation Testing
Eliza Baier	10/11/2024	4.7	Data Creation, Database Testing, Presentation
Ian Johanson	10/11/2024	2	Data Creation, Database Testing
Claranne Fechter	10/16/2024	1	Report/ Presetation
Eliza Baier	10/17/2024	3.1	Report, Database Creation
Ian Johanson	10/17/2024	1.5	Report section draft
Tingting Feng	10/17/2024	3	Report, Presentation
Claranne Fechter	10/17/2024	3.2	Report/ Presetation/ Meeting Prep
Tingting Feng	10/18/2024	1.5	Report, Presentation
Ian Johanson	10/18/2024	3	Report, Presentation
Eliza Baier	10/19/2024	1.5	Report
Claranne Fechter	10/19/2024	1	Report/ Presentation
Claranne Fechter	10/20/2024	1	Practice Presentation
ALL	10/21/2024	1.3	Presentation
Eliza Baier	10/21/2024	1	Presentation
Ian Johanson	10/22/2024	3	Presentation Practice
Tingting Feng	10/22/2024	1	Presentation Practice

DETAILED REQUIREMENTS

Requirement	Details	Status
Enable users to register and customize their profiles	Table needed: Duo_user	Completed
Provide a wide range of courses for users to learn	Table needed: CourseEnrollment	Completed
Organize lessons into different levels to promote step-by-step learning	Tables needed: Lesson, Content_type	Completed
Allow users to track their learning progress, scores, and milestones	Tables needed: Score, Notification	Completed
Include a feature for users to connect with friends and compete in leagues to enhance engagement	Tables needed: Friend, Friendship, Join_league, League	Completed
Offer subscription plans	Tables needed: Subscription_plan, User_plan	Completed
Language certification tests	Tables needed: Test, Test_question, Test_question_info, Test_answer	Completed
Product placement features	Tables needed: Product_placement, Product_placement_inclusion, Client_company	Completed

SQL STATEMENTS - CREATE TABLES

```
-- Create user_plan table
CREATE TABLE "user_plan" (
  "user_plan_id" INTEGER NOT NULL,
  "plan_status" VARCHAR(50) NOT NULL,
  "start_date" DATE,
  "end_date" DATE,
  PRIMARY KEY ("user_plan_id")
);

-- Create duo_user table
CREATE TABLE "duo_user" (
  "user_id" VARCHAR(25) NOT NULL,
  "user_first_name" VARCHAR(100),
  "user_last_name" VARCHAR(100),
  "user_email" VARCHAR(250) NOT NULL,
  "user_age" INTEGER NOT NULL,
  "user_join_date" DATE NOT NULL,
  "user_score" INTEGER NOT NULL,
  "user_native_language" VARCHAR(100) NOT NULL,
  "user_plan_id" INTEGER NOT NULL,
  PRIMARY KEY ("user_id"),
  CONSTRAINT "FK_duo_user.user_plan_id"
    FOREIGN KEY ("user_plan_id")
      REFERENCES "user_plan"("user_plan_id")
);

-- Create course table
CREATE TABLE "course" (
  "course_id" INTEGER NOT NULL,
  "course_name" VARCHAR(200) NOT NULL,
  "course_level" INTEGER NOT NULL,
  "course_percent_progress" DECIMAL NOT NULL,
  PRIMARY KEY ("course_id")
);
```

SQL STATEMENTS - CREATE TABLES

```
-- Create lesson table
CREATE TABLE "lesson" (
  "lesson_id" INTEGER NOT NULL,
  "lanuage_name" VARCHAR(100) NOT NULL,
  "lesson_level" INTEGER NOT NULL,
  "completion_date" DATE,
  "course_id" INTEGER NOT NULL,
  "user_id" VARCHAR(25) NOT NULL,
  PRIMARY KEY ("lesson_id"),
  CONSTRAINT "FK_lesson.user_id"
    FOREIGN KEY ("user_id")
      REFERENCES "duo_user"("user_id"),
  CONSTRAINT "FK_lesson.course_id"
    FOREIGN KEY ("course_id")
      REFERENCES "course"("course_id")
);

-- Create league table
CREATE TABLE "league" (
  "league_id" INTEGER NOT NULL,
  "league_name" VARCHAR(100) NOT NULL,
  "user_rank" INTEGER,
  PRIMARY KEY ("league_id")
);

-- Create score table
CREATE TABLE "score" (
  "score_id" INTEGER NOT NULL,
  "accuracy_percentage" INTEGER NOT NULL,
  "lesson_time" TIME NOT NULL,
  "earned_xp" INTEGER NOT NULL,
  "lesson_id" INTEGER NOT NULL,
  "user_id" VARCHAR(25) NOT NULL,
  "league_id" INTEGER NOT NULL,
```

SQL STATEMENTS - CREATE TABLES

```
PRIMARY KEY ("score_id"),
 CONSTRAINT "FK_score.lesson_id"
 FOREIGN KEY ("lesson_id")
 REFERENCES "lesson"("lesson_id"),
 CONSTRAINT "FK_score.league_id"
 FOREIGN KEY ("league_id")
 REFERENCES "league"("league_id"),
 CONSTRAINT "FK_score.user_id"
 FOREIGN KEY ("user_id")
 REFERENCES "duo_user"("user_id")
);

-- Create friend table
CREATE TABLE "friend" (
 "friend_id" VARCHAR(25),
 "friend_first_name" VARCHAR(100),
 "friend_last_name" VARCHAR(100),
 PRIMARY KEY ("friend_id")
);

-- Create friendship table
CREATE TABLE "friendship" (
 "friend_id" VARCHAR(25) NOT NULL,
 "user_id" VARCHAR(25) NOT NULL,
 "friendship_date" DATE,
 "friendship_time" DATE,
 PRIMARY KEY ("friend_id", "user_id"),
 CONSTRAINT "FK_friendship.user_id"
 FOREIGN KEY ("user_id")
 REFERENCES "duo_user"("user_id"),
 CONSTRAINT "FK_friendship.friend_id"
 FOREIGN KEY ("friend_id")
 REFERENCES "friend"("friend_id")
);
```

SQL STATEMENTS - CREATE TABLES

```
-- Create join_league table
CREATE TABLE "join_league" (
  "user_id" VARCHAR(25) NOT NULL,
  "league_id" INTEGER NOT NULL,
  "join_date" DATE,
  PRIMARY KEY ("user_id", "league_id"),
  CONSTRAINT "FK_join_league.user_id"
    FOREIGN KEY ("user_id")
      REFERENCES "duo_user"("user_id"),
  CONSTRAINT "FK_join_league.league_id"
    FOREIGN KEY ("league_id")
      REFERENCES "league"("league_id")
);

-- Create subscription_plan table
CREATE TABLE "subscription_plan" (
  "plan_id" INTEGER NOT NULL,
  "plan_name" VARCHAR(100) NOT NULL,
  "plan_description" VARCHAR(1000),
  "user_plan_id" INTEGER NOT NULL,
  PRIMARY KEY ("plan_id"),
  CONSTRAINT "FK_subscription_plan.user_plan_id"
    FOREIGN KEY ("user_plan_id")
      REFERENCES "user_plan"("user_plan_id")
);

-- Create test_question table
CREATE TABLE "test_question" (
  "question_id" INTEGER NOT NULL,
  "question_text" VARCHAR(200) NOT NULL,
  "question_type" VARCHAR(200),
  PRIMARY KEY ("question_id")
);
```

SQL STATEMENTS - CREATE TABLES

```
-- Create test_answer table
CREATE TABLE "test_answer" (
  "answer_id" INTEGER NOT NULL,
  "answer_text" VARCHAR(200) NOT NULL,
  "answer_correct" BOOLEAN NOT NULL,
  "question_id" INTEGER NOT NULL,
  PRIMARY KEY ("answer_id"),
  CONSTRAINT "FK_test_answer.quesion_id"
    FOREIGN KEY ("quesion_id")
      REFERENCES "test_question"("question_id")
);

-- Create enrollment table
CREATE TABLE "enrollment" (
  "user_id" VARCHAR(25) NOT NULL,
  "course_id" INTEGER NOT NULL,
  "enrollment_date" DATE,
  "enrollment_time" TIME,
  PRIMARY KEY ("user_id", "course_id"),
  CONSTRAINT "FK_enrollment.user_id"
    FOREIGN KEY ("user_id")
      REFERENCES "duo_user"("user_id"),
  CONSTRAINT "FK_enrollment.course_id"
    FOREIGN KEY ("course_id")
      REFERENCES "course"("course_id")
);

-- Create notification table
CREATE TABLE "notification" (
  "notification_id" INTEGER NOT NULL,
  "notification_title" VARCHAR(100) NOT NULL,
  "notification_text" VARCHAR(1000) NOT NULL,
  "user_id" VARCHAR(25) NOT NULL,
  PRIMARY KEY ("notification_id"),
```

SQL STATEMENTS - CREATE TABLES

```
 CONSTRAINT "FK_notification.user_id"
 FOREIGN KEY ("user_id")
 REFERENCES "duo_user"("user_id")
);

-- Create test table
CREATE TABLE "test" (
  "test_id" INTEGER NOT NULL,
  "test_result" DECIMAL,
  "test_type" VARCHAR(100),
  "user_id" VARCHAR(25) NOT NULL,
  PRIMARY KEY ("test_id")
);

-- Create client company table
CREATE TABLE "client_company" (
  "company_id" INTEGER NOT NULL,
  "company_name" VARCHAR(200),
  PRIMARY KEY ("company_id")
);

-- Create product placement table
CREATE TABLE "product_placement" (
  "product_id" INTEGER NOT NULL,
  "product_name" VARCHAR(200) NOT NULL,
  "product_word" VARCHAR(200) NOT NULL,
  "product_type" VARCHAR(200) NOT NULL,
  "product_country" VARCHAR(200) NOT NULL,
  "incident_price" DECIMAL NOT NULL,
  "company_id" INTEGER NOT NULL,
  PRIMARY KEY ("product_id"),
  CONSTRAINT "FK_product_placement.company_id"
    FOREIGN KEY ("company_id")
      REFERENCES "client_company"("company_id")
);
```

SQL STATEMENTS - CREATE TABLES

```
-- Create product_placement_inclusion table
CREATE TABLE "product_placement_inclusion" (
  "product_id" INTEGER NOT NULL,
  "lesson_id" INTEGER NOT NULL,
  "inclusion_date" DATE,
  PRIMARY KEY ("product_id", "lesson_id"),
  CONSTRAINT "FK_product_placement_inclusion.lesson_id"
    FOREIGN KEY ("lesson_id")
      REFERENCES "lesson"("lesson_id"),
  CONSTRAINT "FK_product_placement_inclusion.product_id"
    FOREIGN KEY ("product_id")
      REFERENCES "product_placement"("product_id")
);

-- Create content_type table
CREATE TABLE "content_type" (
  "content_type_id" INTEGER NOT NULL,
  "content_type_description" VARCHAR(200),
  "lesson_id" INTEGER NOT NULL,
  PRIMARY KEY ("content_type_id"),
  CONSTRAINT "FK_content_type.lesson_id"
    FOREIGN KEY ("lesson_id")
      REFERENCES "lesson"("lesson_id")
);

-- Create test_question_info table
CREATE TABLE "test_question_info" (
  "test_id" INTEGER NOT NULL,
  "question_id" INTEGER NOT NULL,
  PRIMARY KEY ("test_id", "question_id"),
  CONSTRAINT "FK_test_question_info.test_id"
    FOREIGN KEY ("test_id")
      REFERENCES "test"("test_id"),
  CONSTRAINT "FK_test_question_info.question_id"
    FOREIGN KEY ("question_id")
      REFERENCES "test_question"("question_id")
);
```

POPULATED TABLES

client_company | Enter a SQL expression to filter

	company_id	company_name
1	1	LanguageCorp
2	2	EduWorld
3	3	Learnify

content_type | Enter a SQL expression to filter results (use Ctrl+Space)

	content_type_id	content_type_description	lesson_id
1	1	Vocabulary	1
2	2	Grammar	2
3	3	Listening	3
4	4	Speaking	4

course | Enter a SQL expression to filter results (use Ctrl+Space)

	course_id	course_name	course_level	course_percent_progress
1	1	Spanish for Beginners	1	20.5
2	2	French Intermediate	2	45
3	3	German Advanced	3	70
4	4	Italian for Beginners	1	15
5	5	Portuguese Intermediate	2	50
6	6	Japanese Advanced	3	80
7	7	Korean for Beginners	1	25
8	8	Chinese Intermediate	2	60
9	9	Russian Advanced	3	75
10	10	English for Spanish Speakers	1	30

duo_user | Enter a SQL expression to filter results (use Ctrl+Space)

	user_id	user_first_nar	user_last_nam	user_email	user_age	user_join_date	user_score	user_native_language	user_plan_id
1	user001	John	Doe	john.doe@example.com	25	2023-03-01	1,500	English	1
2	user002	Jane	Smith	jane.smith@example.com	30	2023-04-01	2,000	Spanish	10
3	user003	Alice	Johnson	alice.johnson@example.con	28	2022-05-15	1,700	French	2
4	user004	Bob	Brown	bob.brown@example.com	22	2023-06-20	1,200	German	5
5	user005	Charlie	Davis	charlie.davis@example.com	35	2023-07-10	1,900	Italian	3
6	user006	Diana	Miller	diana.miller@example.com	29	2023-08-05	2,100	Portuguese	4
7	user007	Eve	Wilson	eve.wilson@example.com	27	2023-09-01	1,600	Japanese	7
8	user008	Frank	Clark	frank.clark@example.com	32	2023-10-01	1,400	Korean	6
9	user009	Grace	Lee	grace.lee@example.com	24	2023-11-01	1,800	Chinese	9
10	user010	Hank	Martin	hank.martin@example.com	31	2023-12-01	1,550	Russian	8

POPULATED TABLES

enrollment | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text

	user_id	course_id	enrollment_date	enrollment_time
1	user001	1	2023-03-01	09:00:00
2	user002	2	2023-04-01	10:00:00
3	user004	3	2023-06-20	11:00:00
4	user005	1	2023-07-10	12:00:00
5	user006	2	2023-08-05	13:00:00
6	user005	4	2023-09-01	14:00:00
7	user006	5	2023-10-01	15:00:00
8	user007	6	2023-11-01	16:00:00
9	user008	7	2023-12-01	17:00:00
10	user009	8	2023-12-05	18:00:00

friend | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text

	friend_id	friend_first_name	friend_last_name
1	friend001	Michael	Anderson
2	friend002	Nina	Roberts
3	friend003	Oliver	Garcia
4	friend004	Paul	Walker
5	friend005	Samantha	White

friendship | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text

	friend_id	user_id	friendship_date	friendship_time
1	friend001	user001	2023-03-20	2023-03-20
2	friend002	user002	2023-04-25	2023-04-25
3	friend003	user004	2023-06-30	2023-06-30
4	friend004	user005	2023-08-15	2023-08-15
5	friend005	user006	2023-09-10	2023-09-10

POPULATED TABLES

join_league | Enter a SQL expression to filter results (use

	user_id	league_id	join_date
1	user001	1	2023-03-10
2	user002	2	2023-04-15
3	user004	3	2023-06-25
4	user005	1	2023-07-20
5	user006	2	2023-08-15
6	user007	3	2023-11-10
7	user008	1	2023-12-01
8	user009	2	2023-12-10
9	user010	3	2023-12-15

league | Enter a SQL expression to filter results (use Ctrl+Space

	league_id	league_name	user_rank
1	1	Bronze League	1
2	2	Silver League	2
3	3	Gold League	3
4	4	Platinum League	4
5	5	Diamond League	5
6	6	Emerald League	6
7	7	Ruby League	7
8	8	Sapphire League	8
9	9	Amethyst League	9
10	10	Obsidian League	10

lesson | Enter a SQL expression to filter results (use Ctrl+Space)

	lesson_id	lanuage_name	lesson_level	completion_date	course_id	user_id
1	1	Spanish	1	2023-03-15	1	user001
2	2	French	2	2023-04-20	2	user002
3	3	German	3	2023-06-25	3	user004
4	4	Spanish	1	2023-07-15	1	user005
5	5	French	2	2023-08-10	2	user006
6	6	Italian	1	2023-09-15	4	user005
7	7	Portuguese	2	2023-10-10	5	user006
8	8	Japanese	3	2023-11-20	6	user007
9	9	Korean	1	2023-12-05	7	user008
10	10	Chinese	2	2023-12-15	8	user009

POPULATED TABLES

notification | Enter a SQL expression to filter results (use Ctrl+Space)

	notification_id	notification_title	notification_text	user_id
1	1	Lesson Reminder	Don't forget to complete your lesson today!	user001
2	2	New Achievement	You have earned a new badge!	user002
3	3	League Update	You moved up in the league rankings!	user004
4	4	Friend Request	You have a new friend request!	user005
5	5	Course Completion	Congratulations on completing your course!	user006

product_placement | Enter a SQL expression to filter results (use Ctrl+Space)

	product_id	product_name	product_word	product_type	product_country	incident_price	company_id
1	1	Language Book	Libro	Book	Spain	15.99	1
2	2	Flashcards	Cartes	Cards	France	9.99	2
3	3	Vocabulary App	Vokabeln	App	Germany	4.99	3
4	4	Headphones	Auriculares	Electronics	Spain	29.99	1
5	5	Notebook	Cahier	Stationery	France	5.49	2
6	6	Language Guide	Leitfaden	Book	Germany	12.99	3
7	7	Tablet	Tablette	Electronics	France	199.99	2
8	8	Smartphone	Handy	Electronics	Germany	299.99	3
9	9	Bluetooth Speaker	Altavoz	Electronics	Spain	49.99	1
10	10	Grammar Book	Grammatikbuch	Book	Germany	20	3

product_placement_inclusion | Enter a SQL expression to filt

	product_id	lesson_id	inclusion_date
1	1	1	2023-03-15
2	2	2	2023-04-20
3	3	3	2023-06-25
4	4	4	2023-07-15
5	5	5	2023-08-10
6	6	6	2023-09-15
7	7	7	2023-10-10
8	8	8	2023-11-20
9	9	9	2023-12-05
10	10	10	2023-12-15

score | Enter a SQL expression to filter results (use Ctrl+Space)

	score_id	accuracy_percentage	lesson_time	earned_xp	lesson_id	user_id	league_id
1	1	95	00:20:00	150	1	user001	1
2	2	90	00:25:00	200	2	user002	2
3	3	85	00:30:00	250	3	user004	3
4	4	88	00:22:00	180	4	user005	1
5	5	92	00:27:00	220	5	user006	2
6	6	89	00:24:00	190	6	user005	1
7	7	94	00:26:00	210	7	user006	2
8	8	91	00:28:00	230	8	user007	3
9	9	87	00:23:00	170	9	user008	1
10	10	93	00:29:00	240	10	user009	2

POPULATED TABLES

subscription_plan | Enter a SQL expression to filter results (use Ctrl+Space)

	plan_id	plan_name	plan_description	user_plan_id
1	1	Free Plan	Basic access with ads	1
2	2	Plus Plan	Ad-free experience and offline lessons	2
3	3	Family Plan	Access for up to 6 family members	3

test | Enter a SQL expression to filter results (use Ctrl+Space)

	test_id	test_result	test_type	user_id
1	1	85.5	Midterm	user001
2	2	90	Final	user002
3	3	88	Quiz	user004
4	4	92.5	Practice	user006

test_answer | Enter a SQL expression to filter results (use Ctrl+Space)

	answer_id	answer_text	answer_correct	quesion_id
1	1	Hola	[v]	1
2	2	Madrid	[ ]	2
3	3	Paris	[v]	2
4	4	Adieu	[ ]	3
5	5	Au revoir	[v]	3
6	6	4	[v]	4
7	7	5	[ ]	4
8	8	Wie geht es dir?	[v]	5
9	9	Mars	[ ]	6
10	10	Jupiter	[v]	6
11	11	Grazie	[ ]	7
12	12	Thank you	[v]	7
13	13	H2O	[v]	8
14	14	Carbon Dioxide	[ ]	8
15	15	Até logo	[v]	9
16	16	Hasta luego	[ ]	9
17	17	4	[v]	10
18	18	5	[ ]	10
19	19	おはよう	[v]	11
20	20	こんにちは	[ ]	11
21	21	William Shakespeare	[v]	12
22	22	Charles Dickens	[ ]	12
23	23	사랑해요	[v]	13
24	24	고마워요	[ ]	13
25	25	Rome	[v]	14
26	26	Venice	[ ]	14
27	27	Спокойной ночи	[v]	15
28	28	Добрый вечер	[ ]	15

POPULATED TABLES

test_question | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text | Record

	question_id	question_text	question_type
1	1	Translate "Hello" to Spanish	Translation
2	2	What is the capital of France?	Multiple Choice
3	3	Translate "Goodbye" to French	Translation
4	4	What is 2 + 2?	Multiple Choice
5	5	Translate "How are you?" to German	Translation
6	6	What is the largest planet?	Multiple Choice
7	7	Translate "Thank you" to Italian	Translation
8	8	What is the chemical symbol for water?	Multiple Choice
9	9	Translate "See you later" to Portuguese	Translation
10	10	What is the square root of 16?	Multiple Choice
11	11	Translate "Good morning" to Japanese	Translation
12	12	Who wrote "Hamlet"?	Multiple Choice
13	13	Translate "I love you" to Korean	Translation
14	14	What is the capital of Italy?	Multiple Choice
15	15	Translate "Goodnight" to Russian	Translation
16	16	What is 3 * 3?	Multiple Choice
17	17	Translate "Please" to Chinese	Translation
18	18	What is the capital of Japan?	Multiple Choice
19	19	Translate "Excuse me" to French	Translation
20	20	What is the freezing point of water in Celsius?	Multiple Choice
21	21	Translate "Where is the bathroom?" to Spanish	Translation
22	22	Who painted the Mona Lisa?	Multiple Choice
23	23	Translate "Have a nice day" to Italian	Translation
24	24	What is the speed of light?	Multiple Choice
25	25	Translate "See you tomorrow" to German	Translation
26	26	What is the atomic number of oxygen?	Multiple Choice
27	27	Translate "Can you help me?" to Portuguese	Translation
28	28	What is the tallest mountain in the world?	Multiple Choice

POPULATED TABLES

test_question_info | Enter a SQL expre

Grid | Text

	test_id	question_id
1	1	1
2	1	5
3	1	9
4	1	13
5	1	17
6	2	2
7	2	6
8	2	10
9	2	14
10	2	18
11	3	3
12	3	7
13	3	11
14	3	15
15	3	19
16	4	4
17	4	8
18	4	12
19	4	16
20	4	20

user_plan | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text

	user_plan_id	plan_status	start_date	end_date
1	1	Active	2023-01-01	[NULL]
2	2	Inactive	2022-01-01	2022-12-31
3	3	Active	2024-01-01	[NULL]
4	4	Inactive	2022-10-01	2023-10-01
5	5	Active	2024-03-20	[NULL]
6	6	Inactive	2023-07-01	2024-07-01
7	7	Active	2024-04-01	[NULL]
8	8	Inactive	2021-05-01	2022-05-01
9	9	Active	2024-05-01	[NULL]
10	10	Inactive	2023-08-15	2024-02-15

SQL STATEMENTS - SAMPLE QUERIES

```
--Query to list the most popular courses by enrollment
SELECT c.course_name, COUNT(e.user_id) AS total_enrollments
FROM course c
JOIN enrollment e ON c.course_id = e.course_id
GROUP BY c.course_name
ORDER BY total_enrollments DESC;


--Query to Get the Top-Performing Users Based on XP Earned
SELECT u.user_first_name, u.user_last_name, SUM(s.earned_xp) AS total_xp
FROM duo_user u
JOIN score s ON u.user_id = s.user_id
GROUP BY u.user_first_name, u.user_last_name
ORDER BY total_xp DESC
LIMIT 10;


--Query to Find top Revenue by Company
SELECT c.company_name, SUM(p.incident_price) AS total_revenue
FROM product_placement p
JOIN client_company c ON p.company_id = c.company_id
GROUP BY c.company_name
ORDER BY total_revenue DESC
LIMIT 3;


 --Query to find users who have completed a specific course and their scores
SELECT u.user_first_name, u.user_last_name, c.course_name, l.completion_date,
s.accuracy_percentage, s.earned_xp
FROM duo_user u
JOIN lesson l ON u.user_id = l.user_id
JOIN course c ON l.course_id = c.course_id
JOIN score s ON l.lesson_id = s.lesson_id
WHERE l.completion_date IS NOT NULL;
```

SQL STATEMENTS - SAMPLE QUERIES

```
--Query to Find Product Placement in Lessons
SELECT pp.product_name, c.course_name, l.lesson_level, ppi.inclusion_date
FROM product_placement pp
JOIN product_placement_inclusion ppi ON pp.product_id = ppi.product_id
JOIN lesson l ON ppi.lesson_id = l.lesson_id
JOIN course c ON l.course_id = c.course_id
ORDER BY ppi.inclusion_date DESC, l.lesson_level;


--Query to Find Average Test Scores by Course
SELECT c.course_name, AVG(t.test_result) AS average_test_score
FROM course c
JOIN lesson l ON c.course_id = l.course_id
JOIN test t ON l.user_id = t.user_id
GROUP BY c.course_name
ORDER BY average_test_score DESC;
```