**АННОТАЦІЯ**

Ця бакалаврська кваліфікаційна робота призначена для розробки програмного забезпечення для вивчення японської писемності.

Під час роботи над бакалаврською кваліфікаційною роботою було досліджено різноманітні IT-засоби для вивчення японської писемності, виявлено основні переваги і недоліки існуючих засобів, на основі яких було створено вимоги до програмного забезпечення, які було формалізовано у специфікації вимог. Також, для забезпечення належного і правильно функціонування, було проведено тестування розробленого програмного забезпечення за допомогою модульного тестування, а також статичного аналізу коду.

Крім цього, для оцінки економічної доцільності розробленого продукту, було проведено економічний аналіз, який показав, що займатися розробкою цього програмного забезпечення доцільно і допоміг вибрати стратегію для виходу на ринок.

Загальний обсяг роботи: Х сторінок.

**ABSTRACT**

This bachelor qualification work is intended for developing software for learning Japanese written language.

During working on this bachelor qualification work I explored various of IT-tools for written Japanese language learning, revealed main advantages and disadvantages of existing solutions, based on them, was created requirements for software, that was formalized in requirements specification. Also for providing adequate and properly functioning, was performed testing of developed software using unit-tests and static code analysis.

Furthermore, for estimation of economic feasibility of this software, an economic analysis was conducted that shows if developing of this software is advisable. It will also help to choose strategy for entering market.

The total amount of the work are X pages.

**ЗМІСТ**

[ВСТУП 6](#_Toc514060089)

[Розділ 1. Огляд IT – засобів для вивчення японської писемності 7](#_Toc514060090)

[1.1 Огляд предметної області 7](#_Toc514060091)

[1.2 Висновки до розділу аналізу існуючих засобів 11](#_Toc514060092)

[Розділ 2. Постановка завдання для розробки програмноЇ АПЛІКАЦІЇ для ВИВЧЕННЯ ЯПОНСЬКОЇ КАЛІГРАФІЇ 12](#_Toc514060093)

[2.1 Постановка завдання 12](#_Toc514060094)

[2.2 Специфікація вимог до програмного забезпечення для організації комунікації студентів з викладачем на лекції (серверна частина) 13](#_Toc514060095)

[2.2.1 Вступ 13](#_Toc514060096)

[2.2.1.1 Призначення, мета 13](#_Toc514060097)

[2.2.2 Загальний опис 13](#_Toc514060098)

[2.2.2.1 Характеристики продукту 13](#_Toc514060099)

[2.2.2.2 Класи користувачів 13](#_Toc514060100)

[2.2.2.3 Середовище функціонування 14](#_Toc514060101)

[2.2.3 Характеристики системи 14](#_Toc514060102)

[2.2.4 Вимоги до зовнішніх інтерфейсів 18](#_Toc514060103)

[2.2.4.1 Користувацькі інтерфейси 18](#_Toc514060104)

[2.2.4.2 Апаратні інтерфейси 18](#_Toc514060105)

[2.2.4.3 Програмні інтерфейси 19](#_Toc514060106)

[2.2.4.4 Комунікаційні інтерфейси 19](#_Toc514060107)

[2.2.5 Інші нефункційні вимоги 19](#_Toc514060108)

[2.2.5.1 Вимоги продуктивності 19](#_Toc514060109)

[2.3 Висновки до розділу постановка задачі та специфікації вимог 19](#_Toc514060110)

[РОЗДІЛ 3. ПРОЕКТУВАННЯ АРХІТЕКТУРИ ЗАСТОСУВАННЯ 19](#_Toc514060111)

[3.1 Проектування архітектури 19](#_Toc514060112)

[3.2. Вибір інструментальних засобів для реалізації мобільного додатку 22](#_Toc514060113)

[3.3. Проектування класів додатку 23](#_Toc514060114)

[3.4 Проектування бази даних 23](#_Toc514060115)

[3.5 Проектування графічного інтерфейсу користувача 23](#_Toc514060116)

[ДОДАТОК А. Словник предметної області 23](#_Toc514060117)

# **ВСТУП**

Японія – передова країна, яка щороку дивує людство новими винаходами. Її своєрідна культура захоплює міліони людей по всьому світу: кабукі, японська кухня, аніме, ікебана, манга та багато інших особливостей цього народу змушують людей ставати на тернистий шлях вивчення японської мови.

Насправді, граматика мови не є складною. Найбільшою проблемою, з якою стикаються люди, що самостійно вивчають японську мову – це її писемність. Дві абетки і ще близько 3000 активно використовуваних ієрогліфів кандзі, запозичених з Китаю можуть просто заплутати.

Окрім того, є проблема систематизації та пріоритизації вивчення писемності. Не зрозуміло, які ієрогліфи потрібно вивчати в першу чергу. До того ж, більшість ієрогліфів мають кілька варіантів прочитання, що робить задачу ще складнішою. А без систематичних занять писемність вивчити абсолютно неможливо, адже схожі між собою ієрогліфи, однакові прочитання для кількох різних ієрогліфів і різні варіанти прочитання ієрогліфів всередині слів забуваються без практики.

Отож, метою цієї бакалаврської роботи є розробити програмне забезпечення, що спростило б вивчення японської писемності, а саме надавало рекомендації щодо порядку вивчення, надавало б флеш-картки для інтервального повторення, а також контролювало б процес інтервального повторення за допомогою нагадувань.

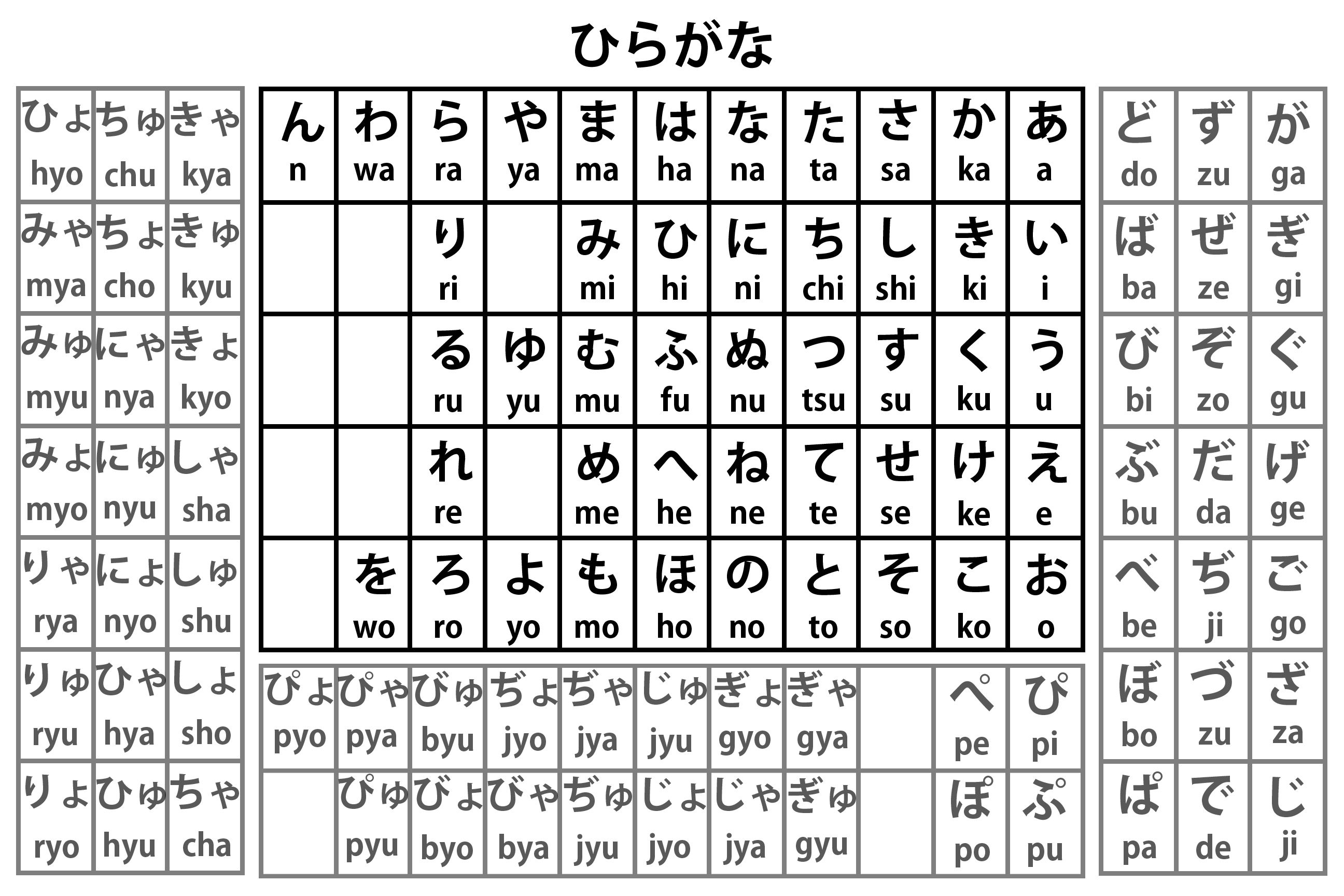
**Розділ 1. Огляд IT – засобів для вивчення японської писемності**

# **1.1 Огляд предметної області**

Вивчення японських ієрогліфів зазвичай починають із двох абеток – хірагани та катакани, адже усі кандзі можна записати ними. Кожна з них містить 48 знаків, кожен з яких позначає один склад або мору.

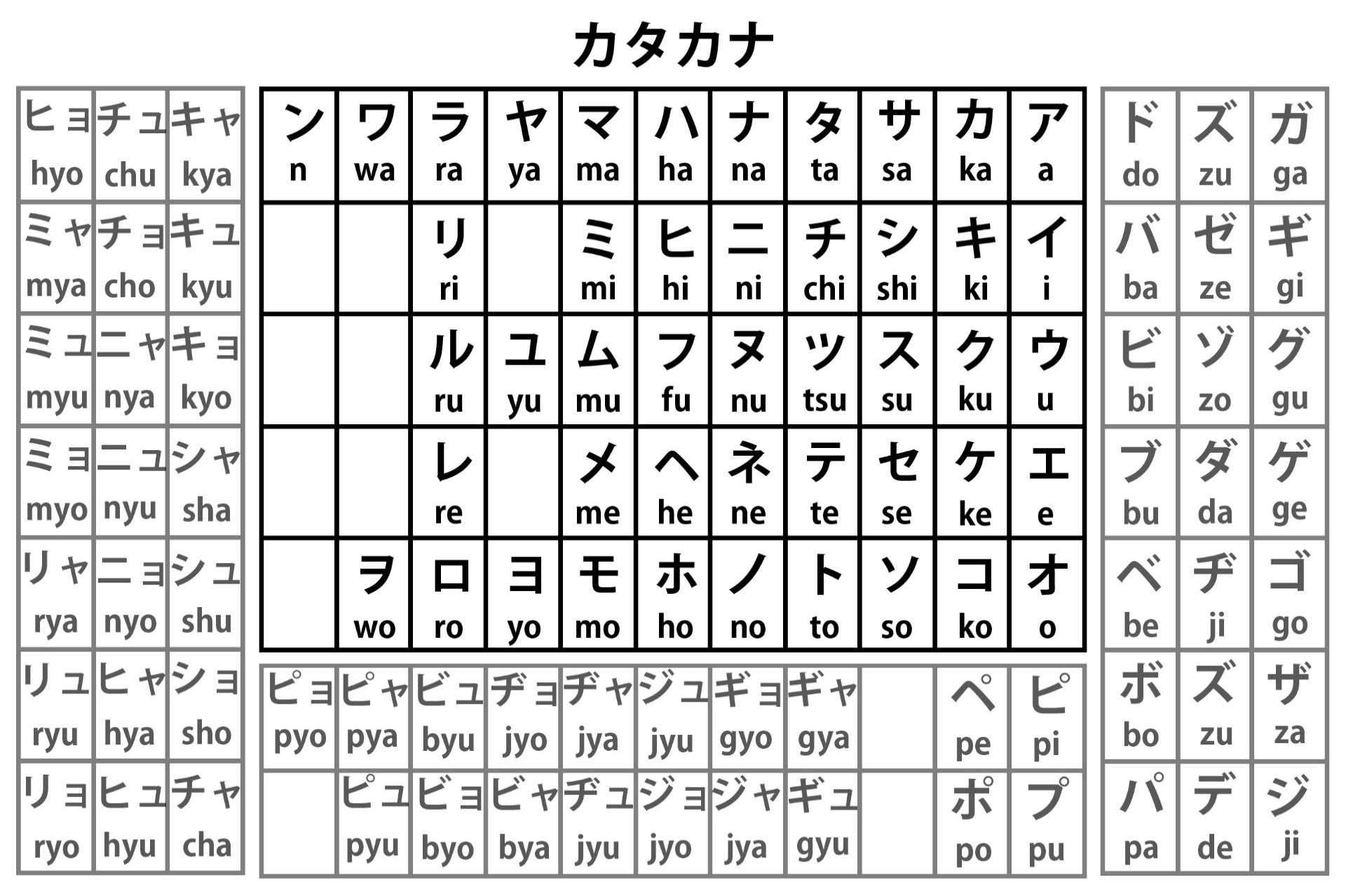
Хірагана (табл. 1.1.1) застосовується для запису граматичних часток, відмінкових суфіксів, післякорінних змінюваних частин слова, фонетичних підказок для ієрогліфів.

Таблиця 1.1.1. Знаки хірагани



Катакана (табл.1.1.2) сьогодні використовується переважно для запису запозичених слів європейського походження, іншомовних власних назв (країни, імена тощо), та термінів, еквіваленти яких відсутні у японській мові.

Таблиця 1.1.2. Знаки катакани



Ієрогліфи кандзі супроводжуються в навчальній літературі допоміжною фуріґаною, що допомагає читати кандзі за допомогою хірагани. Це створено для того, аби людина, що не знає, як читається та чи інша кандзі, могла прочитати і зрозуміти слова, що ці кандзі означають. Таким чином створена навчальна література для японських дітей, адже ієрогліфи кандзі вивчаються ними в молодшій школі (1006 ієрогліфів) та старшій школі (1130 кандзі), щороку від 80 і до 200. Виглядає це так:

Рис. 1.1.1. Ієрогліф «чоловічий» із пояснювальною фуріґаною «отоко»

Для іноземців, що вивчають японську мову складено спеціальний тест – «Ніхонго Норьоку Шікен». Цей тест розподілений на 5 рівнів (N5 – найпростіший, N1 – найскладніший), для здачі кожного з яких потрібно володіти певним переліком кандзі. Відповідно, саме цей порядок вивчення рекомендується японською асоціацією допомоги міжнародної освіти, що розробила тести для цих рівнів. Ці тести проводяться 1-2 рази на рік (в Україні – 1 раз на рік, в Києві).

Ієрогліфи кандзі складаються із радикалів. В японській мові існує 214 радикалів, кожен із яких має своє значення. Знання радикалів дуже допомагає при вивченні кандзі, адже радикали позначають, до якої частини мови відноситься ієроглів, його характеристичне забарвлення, тощо, через що подекуди значення ієрогліфа можна вгадати, розуміючи його складові радикали (рис.1.1.2).

森

Рис.1.1.2. Ієрогліф «Ліс», що складається з трьох ієрогліфів «Дерево»

На основі отриманої інформації проведемо аналіз застосувань, що призначені суто для вивчення японської каліграфії. Для розгляду було обрано мобільний додаток «Japanese Characters». Його перевагами є:

* Є можливість вивчення як кандзі, так і хірагани та катакани;
* В додатку є чітка послідовність, за якою потрібно вивчати ієрогліфи. Вони поділені на 6 груп, кожна з яких поділена на уроки;
* В групах є вправи на перевірку вивченого;
* В кожному уроці є флеш-картки для усіх ієрогліфів з уроку;
* Можливість переглянути увесь список ієрогліфів в уроці;
* Статистика часу навчання;
* Історія пройдених уроків.

Серед недоліків аплікації було виявлено такі:

* Відсутність нагадувань про перегляд карток (флеш-картки без системи Лейтнера не є дуже ефективними);
* Відсутність пошуку ієрогліфів;
* Не має можливості вивчати радикали;
* Деякий функціонал доступний лише у платній версії (прочитання кандзі хіраганою; демонстрація порядку написання; збереження улюблених кандзі);
* Реклама.

Також, для кращого розуміння предметної області варто пояснити систему Лейтнера. Ця методика використовує особливості нашого мозку, що дозволяє з меншими зусиллями запам'ятовувати інформацію, використовуючи перевірки та повтори. В цій системі використовуються флеш-картки, що представляють собою двосторонню картку, одна сторона якої – це термін, який потрібно запам'ятати, а зворотня – пояснення до нього. Наприклад, у випадку з ієрогліфами, на одній стороні буде сам ієрогліф, а на іншій – його прочитання або переклад. Процес навчання за допомогою флеш-карток та системи Лейтнера виглядає таким чином:

Учень бере картку, повернуту терміном до себе. Якщо він знає значення, то він перекладає термін у другий стос. Якщо ні, то вертає його в першу групу. Процес повторюється, допоки учень не перегляне усі картки. Після того робиться перерва в навчанні. Термін вважається засвоєним, якщо через кілька інтервальних повторів картка попадає в останній стос (зазвичай – четвертий). Якщо учень не знає терміну, в якому стосі він би не був, картка повертається в першу групу.

Особливість такого підходу у вивченні полягає в тому, що інформація, яка регулярно появляється в тимчасовій пам’яті, зберігається в глибинах пам’яті, і з кожною наступною згадкою пам’ять про неї буде міцнішати, допоки інформація не буде збережена у довгочасну пам’ять.

Як приклад програми, що надає можливість для навчання за допомогою флеш-карток, було розглянуто мобільний додаток «Quizlet», що надає флеш-картки для різних напрямків навчання. Отож, такий додаток надає користувачу такі можливості:

* Пошук флеш-карток на різну тематику;
* Створення власних флеш-карток (вони розміщуються для загального доступу);
* Перегляд усіх можливих наборів карток;
* Перегляд карток у наборах;
* Практика у вивченні карток, де потрібно вгадати, що відповідає визначенню на картці;
* 3 різні гри для вивчення:
  + Написання визначення для картки;
  + Співставлення визначення і завдання картки;
  + Тест із різними варіантами питань, де лише один варіант вірний.
* Можливість прослухати завдання і визначення картки;
* Збереження набору карток для вивчення в офлайн режимі;
* Збереження окремих карток в категорію улюблених.

Ця аплікація є більш загальною і підходить не лише для вивчення японської писемності, але й для вивчення будь-чого у будь-якій сфері. Проте, через цю універсальність, аплікація має ряд недоліків:

* Відсутня чітка схема для навчання;
* Відсутність нагадувань про перегляд карток (флеш-картки без системи Лейтнера не є дуже ефективними);
* Оскільки самі набори карток створюються користувачами, вони не є найдійним джерелом, бо можуть містити помилки;
* Безліч наборів, що містять ті самі, або схожі дані;
* Не можливо навчатися, використовуючи кілька наборів водночас.

# **1.2 Висновки до розділу аналізу існуючих засобів**

Отож, було проведено аналіз існуючих засобів для навчання за методом Лейтнера, а також додатків для вивчення японських ієрогліфів. Ці два види програм було б варто об’єднати, аби полегшити процес вивчення японської писемності. Програми ці мають ряд переваг, які варто було б включити в кращу версію додатку, та недоліків, яких варто уникнути.

# **Розділ 2. Постановка завдання для розробки програмноЇ АПЛІКАЦІЇ для ВИВЧЕННЯ ЯПОНСЬКОЇ КАЛІГРАФІЇ**

# **2.1 Постановка завдання**

На основі опису предметної області, сформовано таке завдання: необхідно розробити мобільний додаток, що надаватиме змогу навчатися японської писемності за допомогою системи Лейтнера і буде позбавлений недоліків існуючих рішень.

Отож, найважливішими завданнями такого додатку буде надання структурованої інформації про ієрогліфи з порядком їх вивчення, а також контроль навчання за системою Лейтнера.

Для зручності користувача варто забезпечити його додатковими функціями, як от можливість практики написання ієрогліфів, бо при вивченні ієрогліфів важливим є порядок написання рисок в ієрогліфі. До того ж, практика в написанні ієрогліфів перевіряє активне володіння ієрогліфом.

Також зручною була б наявність в системі, поруч з ієрогліфами, радикалів, адже вивчення радикалів полегшує вивчення та розуміння кандзі.

Корисним було б забезпечити користувача пошуком, адже при ознайомленні із японською літературою, людина, що навчається, часто зустрічатиме ієрогліфи, які їй ще невідомі. Така функція дозволить знайти прямий переклад ієрогліфа, а також його можливі прочитання.

Вхідні дані для роботи: літературні джерела для вивчення японської писемності та мови, документація по мові програмування Java, документація до платформи Android, документація до бібліотеки KanjiVG, специфікація вимог.

Вихідні дані: мобільний додаток, який реалізовує поставлену задачу.

# **2.2 Специфікація вимог до програмного забезпечення для організації комунікації студентів з викладачем на лекції (серверна частина)**

# **2.2.1 Вступ**

# **2.2.1.1 Призначення, мета**

Мета даного проекту – це розробка мобільного додатку для полегшення традиційного способу вивчення японських ієрогліфів за допомогою системи Лейтнера.

# **2.2.2 Загальний опис**

# **2.2.2.1 Характеристики продукту**

У загальному, в даному мобільному додатку можна виділити такі особливості:

* Користувач може переглядати ієрогліфи згруповані на групи за рівнем, а також абетки кана та радикали;
* Користувач може переглядати прочитання, переклад і процес написання ієрогліфа;
* Користувач може практикувати написання ієрогліфів, з урахуванням порядку написання рисок;
* Користувач отримує нагадування про необхідність повторення вивчених раніше ієрогліфів;
* Користувач може бачити свій прогрес по вивченню, скількома ієрогліфами він володіє, скільки ієрогліфів є в кожній із груп вивчення.

# **2.2.2.2 Класи користувачів**

Даний додаток міститиме усього один клас користувача – учень, що використовує програму для вивчення ієрогліфів. Усі дані аплікації будуть надані у внутрішній базі даних і наповнюватимуться лише єдиний раз і не потребуватимуть частих змін, тому роль адміністратора не буде потрібна. Якщо ж відбуватимуться зміни у правописі ієрогліфів, вони будуть надані зовнішнім джерелом (бібліотека KanjiVG).

# **2.2.2.3 Середовище функціонування**

Мобільний додаток для роботи вимагає операційну систему Android версії 5.0.0 (Lollipop) чи старше.

# **2.2.3 Характеристики системи**

1. **Перегляд радикалів**
   * 1. Опис і пріоритет  
        Пріоритет – середній. Можливість переглядати радикали – складові ієрогліфи кандзі.
     2. Послідовність дія-відгук
2. Користувач відкриває сторінку, що містить список радикалів;
3. Користувач може переглянути список радикалів із їх прочитанням, кількістю рисок в ньому, значенню.
   * 1. Функціональні вимоги

REQ 1. Відображення коректного списку радикалів із їх прочитаннями та значеннями.

1. **Перегляд абеток кана**
   * 1. Опис і пріоритет  
        Пріоритет – високий. Можливість переглядати символи абеток хірагана і катакана.
     2. Послідовність дія-відгук
2. Користувач відкриває сторінку, що містить список символів абетки кана;
3. Користувач переглядає їх, отримуючи як інформацію про вигляд символу, так і його прочитання латиною;
4. Користувач може перемикнути перегляд абетки катакани, на перегляд хірагани і навпаки за допомогою спеціальної кнопки.
   * 1. Функціональні вимоги

REQ 1. Абетка повинна відображатися коректно і містити усі символи разом із їх прочитаннями;

REQ 2. Можливість перемикання між абетками катакана і хірагана.

1. **Перегляд ієрогліфів кандзі**
   * 1. Опис і пріоритет  
        Пріоритет – високий. Можливість переглядати символи кандзі погруповані за рівнем вивчення в японських школах.
     2. Послідовність дія-відгук
2. Користувач відкриває сторінку, що містить список рівнів вивчення, від N5 (найлегші) до N1 (найскладніші);
3. Користувач обирає в списку рівень, до якого хоче вивчати ієрогліфи, натискаючи на нього;
4. Система переводить користувача на сторінку зі списком усіх ієрогліфів, що стосуються даного рівня. Ієрогліфи містять інформацію про прочитання, значення.
   * 1. Функціональні вимоги

REQ 1. Кандзі повинні відображатися коректно відповідно до рівня, на якому ними потрібно володіти;

REQ 2. Поруч з інформацією про символ кандзі повинні міститися варіанти прочитання, значення цього ієрогліфа.

1. **Перегляд написання ієрогліфа** 
   * 1. Опис і пріоритет  
        Пріоритет – середній. Можливість переглядати порядок і напрямок написання окремих рисок ієрогліфів.
     2. Послідовність дія-відгук
2. Користувач відкриває сторінку, що містить список символів абетки кана або сторінку із ієрогліфами кандзі або сторінку з радикалами;
3. Користувач натискає на ієрогліф, який хоче переглянути;
4. Система переводить користувача на сторінку із великим зображенням ієрогліфа, що анімує порядок його написання.
   * 1. Функціональні вимоги

REQ 1. Анімація обраного ієрогліфа (як кандзі, так і символів хірагани та катакани та радикалів) з врахуванням порядку і напрямку написання рисок;

REQ 2. Можливість повернутись до сторінки, з якої було здійснено перехід на перегляд анімації обраного ієрогліфу.

1. **Перегляд флеш карток ієрогліфів**
   * 1. Опис і пріоритет  
        Пріоритет – високий. Можливість переглядати ієрогіфи у форматі флеш карток.
     2. Послідовність дія-відгук
2. Користувач відкриває сторінку, що містить список символів абетки кана або сторінку із ієрогліфами кандзі або сторінку з радикалами;
3. Користувач натискає на кнопку переглядати;
4. Система переводить користувача на сторінку із великим зображенням ієрогліфа та стрілками вліво та вправо для перемикання між ієрогліфами; При натисненні на ієрогліф відображається його значення, прочитання онйомі і кунйомі. При натисненні на визначення ієрогліфа повторно відображається ієрогліф. Для ієрогліфів, які користувач ще не знає внизу екрану є кнопка «Пам’ятаю».
5. Зверху екрану є кнопка, за допомогою якої користувач може відфільтрувати флеш картки по рівням.
   * 1. Функціональні вимоги

REQ 1. Коректне співставлення ієрогліфа, його визначення та прочитання;

REQ 2. Можливість відфільтрувати ієрогліфи тих рівнів, які користувач ще не знає.

1. **Нагадування про необхідність повторення ієрогліфів**
   * 1. Опис і пріоритет  
        Пріоритет – високий. Висвітлення нагадування при не активному додатку у центрі повідомлень ОС Android.
     2. Послідовність дія-відгук
2. Користувач активовує телефон і бачить повідомлення про необхідність повторення ієрогліфів.
   * 1. Функціональні вимоги

REQ 1. Повідомлення можна деактивувати;

REQ 2. Повідомлення відображається лише тоді, коли хоча б для 1 ієрогліфа пройшов відлік часу від останнього повторення.

1. **Перегляд рівня вивчення різних ієрогліфів**
   * 1. Опис і пріоритет  
        Пріоритет – низький. Можливість окремо переглядати ієрогліфи в кожному з рівнів пам’яті, що означають певне володіння.
     2. Послідовність дія-відгук
2. Користувач відкриває сторінку флеш карток;
3. Система надає користувачеві розгорнутий список рівнів того, наскільки користувач пам’ятає певні ієрогліфи.
   * 1. Функціональні вимоги

REQ 1. Список пам’яті відповідає тому, що було попередньо введенне користувачем;

REQ 2. Можливість згортати підсписки по рівням пам’яті, а також по групах (радикали, кана, кандзі з рівнями N5-N1).

1. **Вивчення ієрогліфів за системою Лейтнера**
   * 1. Опис і пріоритет  
        Пріоритет – високий. Групування ієрогліфів за рівнем вивчення їх користувачем.
     2. Послідовність дія-відгук
2. Користувач відкриває сторінку, що містить список символів абетки кана або сторінку із ієрогліфами кандзі;
3. Користувач натискає кнопку «Перевірити»;
4. Користувач переходить на сторінку, що відображатиме флеш картки ієрогліфів, що користувач знає хочаб на якомусь рівні по-одному великим розміром на екрані. При натисненні на ієрогліф відображається його значення, прочитання онйомі і кунйомі. При натисненні на визначення ієрогліфа повторно відображається ієрогліф. Внизу екрану є кнопки «Пам’ятаю» і «Забув».
5. При натисненні кнопки «Пам’ятаю», ієрогліф переходить в групу ієрогліфів, які користувач пам’ятає на найгіршому рівні і починається відлік часу до наступного повторення цього ієрогліфа.
   * 1. Функціональні вимоги

REQ 1. Можливість перегляду флеш карток ієрогліфів;

REQ 2. Наявність 5 груп вивчення (невідомий, базовий, середній, добрий, вивчений);

REQ 3. Переведення запам'ятованих ієрогліфів в наступну групу по рівню вивчення;

REQ 4. Переведення забутих ієрогліфів в групу базових;

REQ 5. Надання в тесті лише тих ієрогліфів, які пора повторити;

REQ 6. Кнопка «Перевірити» не активна, якщо нема що повторяти.

# **2.2.4 Вимоги до зовнішніх інтерфейсів**

# **2.2.4.1 Користувацькі інтерфейси**

Користувач може взаємодіяти з мобільною аплікацією лише за допомогою сенсорного екрану.

# **2.2.4.2 Апаратні інтерфейси**

Аплікація незалежна від акселерометра, орієнтація програми не змінюється з поворотом екрану. Інші апаратні інтерфейси не впливають на роботу системи.

# **2.2.4.3 Програмні інтерфейси**

Доступ до бази даних (SQLite) відбуватиметься за допомогою нативної бібліотеки платформи Android – android.database.sqlitte.

# **2.2.4.4 Комунікаційні інтерфейси**

Аплікація не використовує ніяких комунікаційних інтерфейсів.

# **2.2.5 Інші нефункційні вимоги**

# **2.2.5.1 Вимоги продуктивності**

Мобільний додаток повинен бути чутливим до дій користувачів і миттєво на них відповідати. Уся інформація про ієрогліфи повинна завантажуватись не довше ніж за 1 секунду.

**2.2.5.2 Вимоги безпеки**

Додаток не використовує персональних даних користувачів.

**2.2.5.3 Атрибути якості програмного продукту**

* Адаптивність до різних девайсів;
* Стійкість до помилок;
* Зручність використання;
* Легкість підтримки.

2.3 Висновки до розділу постановка задачі та специфікації вимог

Як підсумок розділу, було розроблено специфікацію усіх функцій та вимог. У розділі було сформоване завдання на розробку мобільного додатку, а також описані вхідні та вихідні дані для розробки мобільної аплікації.

# **РОЗДІЛ 3. ПРОЕКТУВАННЯ АРХІТЕКТУРИ ЗАСТОСУВАННЯ**

# **3.1 Проектування архітектури**

Оскільки аплікація розробляється для платформи Android, є сенс скористатись рекомендаціями щодо розробки під цю платформу від компанії-засновника – Google. Отож, стандартним рішенням для розробки додатку для Android є модель, що зображена на рис. 3.1. Це – архітектурний шаблон Model-View-ViewModel.



Рис. 3.1. Типова модель Android-додатку

Ця архітектура має такі складові:

* Activity – візуальний компонент додатку. Він виводить на екран дані, з якими вже може взаємодіяти користувач. Зазвичай, Activity займає весь екран. В коді репрезентується користувацькими класами, що наслідують клас Activity.
* ViewModel – прошарок між бізнес-логікою та представленням. Існує, аби надавати дані до Activity від репозиторіїв. Як тільки дані змінюються, вони тут же будуть доправлені до шару репрезентації. Ця компонента також захищає додаток від втрати даних у разі повороту екрану. В коді репрезентується класами-наслідниками ViewModel.
* Repository – прошарок між логікою додатку та базою даних. Дозволяє легко отримувати дані з бази у потрібній репрезентації. Він дозволяє взаємодіяти як з внутрішньою, так і зовнішньою базою даних.
* Model – представлення даних із бази даних. Так зване відображення, або мапінг моделі даних. Власне, ці дані оброблятимуться в бізнес-логіці додатку. У випадку класичного підходу мапінг відбувається автоматично за допомогою бібліотеки Room, що служить абстракцією над прямою взаємодією з базою через SQLite.
* Remote data source – віддалене джерело даних. Зазвичай веб-сервіс, що надає зовнішній доступ за певним API. У випадку роботи з таким сервісом потрібно враховувати можливі затримки в передачі даних і будувати систему так, аби ці затримки не впливали на користувача.

Цей архітектурний патерн підходить з певними правками. Оскільки дані в додатку не можуть змінюватись незалежно від користувача, тобто дані в базі не можуть змінюватись ззовні і відповідно потреба в live-заміні даних в користувацькому інтерфейсі відсутня, можна замінити ViewModel компонент на шар Presenter, що не замінятиме дані при зміні даних в базі даних, а лише при відповідному запиті презентаційного шару. Також, в аплікації відсутня потреба доступу до віддаленої бази даних, тому цей компонент можна взагалі забрати. Отож, фінальний вигляд архітектури матиме такий вигляд, як на рис. 3.2.



Рис. 3.2. Вигляд обраної архітектури додатку

Також при побудові додатку варто врахувати особливості життєвого циклу Android Activity, що зображений на рис. 3.4. Життєвий цикл тісно пов'язаний із взаємодією користувача із додатком, то ж його варто враховувати якщо, наприклад, користувач згортає додаток.

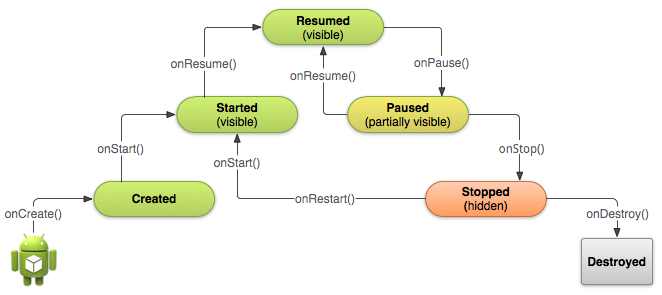


Рис. 3.4. Життєвий цикл Activity

# **3.2. Вибір інструментальних засобів для реалізації мобільного додатку**

Додаток розробляється для платформи Android, отож доцільним є використання мови Java для розробки.

Процес розробки відбуватиметься на платформі Android Studio, що дає найкращий набір інструментів для розробки мобільних додатків, а також усі переваги платформи Intellij IDEA, на базі якої він розроблений (як от, наприклад, IntelliSence).

Для побудови додатку, розв’язування зовнішніх залежностей буде використаний Build-засіб Gradle. Скрипти для цього засобу пишуться на Groovy, за рахунок чого він гнучкіший за XML-базовані build-засоби (як от Maven чи Ant).

Базою даних, що міститиме усі дані, буде SQLite. Обрано саме цю базу, адже вона є локальною базою за замовчуванням для проектів Android. Віддаленої бази не потрібно. Доступ до локальної бази відбуватиметься через бібліотеку Room, що виступає шаром абстракції над SQLite і дозволяє з легкістю відображати дані бази даних на класи Java.

Візуальна частина аплікації буде описана за допомогою стандартних XML файлів. Ці файли зв’язуються із Activity, яку вони репрезентують. Вони використовують свій внутрішній синтаксис для візуальних компонентів. Через те, що розробка відбуватиметься в середовищі Android Studio, не варто турбуватись через синтаксис, бо валідація його відбуватиметься ще до компіляції додатку.

Відображення порядку написання рисок ієрогліфів відбуватиметься за допомогою бібліотеки KanjiVG, що надає SVG-файли для усіх відомих ієрогліфів японської мови (не лише кандзі, як може здатись з назви, але й радикалів та кани). Ці файли містять всі необхідні дані для створення анімації написання ієрогліфа, як от складові радикали і порядок рисок.

# **3.3. Проектування класів додатку**

Основними класами системи розробленої системи є:

Діаграму класів можна побачити в додатку B.

3.4 Проектування бази даних

Робота із базою

Схему бази даних можна побачити в додатку С.

# **3.5 Проектування графічного інтерфейсу користувача**

# **ДОДАТОК А. Словник предметної області**

* Аніме – японська анімація. Головна відмінність від анімації інших країн полягає в тому, що аніме орієнтується не на дитячу, а переважно на підліткову чи навіть дорослу аудиторії.
* Ікебана – традиційне мистецтво аранжування квітів у Японії.
* Кабукі – один з видів традиційного театру Японії.
* Кандзі – ієрогліфічне письмо, складова частина японської писемності. Запозичене японцями в Китаї у 5-6 століттях. До запозичених знаків були додані ієрогліфи, розроблені самими японцями.
* Катакана – японська складова абетка кана. Її знаки походять від складових частин китайських ієрогліфів. Сьогодні використовується переважно для запису запозичених слів європейського походження, іншомовних власних назв (країни, імена тощо), та термінів, еквіваленти яких відсутні у японській мові.
* Кунйомі – японське прочитання, яка засноване на вимові споконвічних японських слів, до яких були за змістом підібрані китайські ієрогліфи. Інакше кажучи, кунйомі — це переклад китайського знаку японською мовою.
* Манга – японський різновид коміксів.
* Онйомі – китайсько-японське прочитання або японська інтерпретація китайської вимови ієрогліфа.
* Радикал - ієрогліф, який є складовою частиною іншого ієрогліфа і який часто дозволяє пояснити його значення (тобто є семантичним елементом).
* Система Лейтнера — широко використовуваний метод для ефективного запам’ятовування та повторення за допомогою флеш-карток, запропонований німецьким вченим і журналістом Себастьяном Лейтнером. Ця система — просте застосування принципу інтервальних повторювань, коли картки повторюються щоразу через більші проміжки часу.
* Фуріґана - фонетичні підказки в японській мові. Складаються з маленьких знаків кани, що надруковані поруч із ієрогліфами або іншими символами, і вказують на їх прочитання.
* Флеш картки – це навчальні картки, що містять запитання на одній сторона, а на іншій – відповідь до нього. Часто використовуються для інформації, яку потрібно регулярно повторювати, аби запам’ятати.
* Хірагана – японська складова абетка кана. Використовується для запису граматичних часток, відмінкових суфіксів, післякорінних змінюваних частин слова, фонетичних підказок для ієрогліфів, а також дошкільній літературі.