|  |  |
| --- | --- |
| **Politechnika Świętokrzyska w Kielcach**  **Wydział Elektrotechniki, Automatyki i Informatyki** | |
| **Symulacja komputerowa** | |
| Projekt – symulacja komputerowa | |
| Mateusz Bonar  1ID21A  Wiktor Wójcik  1ID21B |  |

**Spis treści**

[1. Wstęp teoretyczny 3](#_Toc73905811)

[1.1. Android studio 3](#_Toc73905812)

[1.2. Gradle 4](#_Toc73905813)

[1.3. Java 4](#_Toc73905814)

[2. Założenia projektowe 5](#_Toc73905815)

[3. Przedstawienie projektu 7](#_Toc73905816)

[4. Część praktyczna 8](#_Toc73905817)

[5. Podsumowanie 10](#_Toc73905818)

# Wstęp teoretyczny

Na potrzeby projektu przygotowano aplikację pozwalającą na wykonywanie reakcji chemicznych. Do pracy z projektem wykorzystano:

## Android studio

Android Studio to oficjalne zintegrowane środowisko programistyczne (IDE) zaprojektowany specjalnie dla rozwoju Android. Jest dostępny do pobrania w systemach operacyjnych Windows , macOS i Linux lub jako usługa oparta na subskrypcji w 2020 r. Jest zamiennikiem dla Eclipse Android Development Tools (E-ADT) jako podstawowe IDE do tworzenia natywnych aplikacji na Androida.

Android Studio zostało ogłoszone 16 maja 2013 roku na konferencji Google I/O . Był w fazie wczesnego dostępu, począwszy od wersji 0.1 w maju 2013 roku, a następnie wszedł w fazę beta, począwszy od wersji 0.8, która została wydana w czerwcu 2014 roku. Pierwsza stabilna wersja została wydana w grudniu 2014 roku, począwszy od wersji 1.0.

7 maja 2019 r. Kotlin zastąpił Javę jako preferowany język Google do tworzenia aplikacji na Androida. Java jest nadal obsługiwana, podobnie jak C++ . Specyficzną cechą Android Studio jest brak możliwości wyłączenia funkcji autozapisu.

W obecnej stabilnej wersji dostępne są następujące funkcje:

* Wsparcie kompilacji oparte na Gradle
* Refaktoryzacja i szybkie poprawki dla Androida
* Narzędzia lintowe do wyłapywania wydajności, użyteczności, kompatybilności wersji i innych problemów
* Integracja ProGuard i możliwości podpisywania aplikacji
* Oparte na szablonach kreatory do tworzenia popularnych projektów i komponentów systemu Android
* Bogaty edytor układów, który umożliwia użytkownikom przeciąganie i upuszczanie komponentów interfejsu użytkownika, możliwość podglądu układów na wielu konfiguracjach ekranów
* Obsługa tworzenia aplikacji Android Wear
* Wbudowana obsługa Google Cloud Platform, umożliwiająca integrację z Firebase Cloud Messaging (wcześniej „Google Cloud Messaging”) i Google App Engine
* Android Virtual Device (Emulator) do uruchamiania i debugowania aplikacji w Android Studio.

## Gradle

Gradle to narzędzie do automatyzacji kompilacji do tworzenia oprogramowania w wielu językach. Kontroluje proces rozwoju w zadaniach kompilacji i pakowania do testowania, wdrażania i publikowania. Obsługiwane języki obejmują Java ( Kotlin , Groovy , Scala ), C / C++ i JavaScript .

Gradle opiera się na koncepcjach Apache Ant i Apache Maven i wprowadza język specyficzny dla domeny oparty na Groovy i Kotlin , w przeciwieństwie do konfiguracji projektu opartej na XML , używanej przez Mavena. Gradle używa ukierunkowanego grafu acyklicznego do określenia kolejności, w jakiej mogą być uruchamiane zadania, poprzez zapewnienie zarządzania zależnościami. Gradle jest dystrybuowane jako oprogramowanie typu open source na licencji Apache License 2.0 i zostało wydane po raz pierwszy w 2007 roku.

## Java

Java to zestaw oprogramowania komputerowego i specyfikacji opracowanych przez Jamesa Goslinga w Sun Microsystems , który został później przejęty przez Oracle Corporation , który zapewnia system do tworzenia oprogramowania aplikacji i wdrażania go w wieloplatformowym środowisku obliczeniowym. Java jest używana w wielu różnych platformach obliczeniowych, od urządzeń wbudowanych i telefonów komórkowych po serwery i superkomputery dla przedsiębiorstw. Aplety Java, które są mniej powszechne niż samodzielne aplikacje Java, były zwykle uruchamiane w bezpiecznych, środowiska piaskownicy, aby zapewnić wiele funkcji natywnych aplikacji dzięki osadzaniu na stronach HTML .

Pisanie w języku programowania Java jest podstawowym sposobem tworzenia kodu, który zostanie wdrożony jako kod bajtowy w wirtualnej maszynie Java (JVM); Kompilatory kodu bajtowego są również dostępne dla innych języków, w tym Ada , JavaScript , Python i Ruby. Ponadto zaprojektowano kilka języków do natywnego działania na JVM, w tym Clojure, Groovy i Scala. Składnia Javy mocno zapożycza się z C i C++ , ale funkcje zorientowane obiektowo są wzorowane na Smalltalki Cel-C. Java unika pewnych konstrukcji niskiego poziomu, takich jak wskaźniki i ma bardzo prosty model pamięci, w którym obiekty są alokowane na stosie (podczas gdy niektóre implementacje, np. wszystkie obecnie obsługiwane przez Oracle, mogą zamiast tego używać optymalizacji analizy ucieczki do alokacji na stosie) a wszystkie zmienne typów obiektów są referencjami. Zarządzanie pamięcią odbywa się za pomocą zintegrowanego automatycznego wyrzucania śmieci, wykonywanego przez JVM.

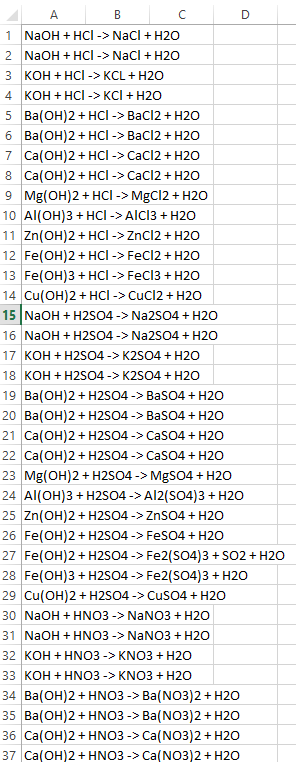
# Założenia projektowe

Założeniem projektowym było przygotowanie rozwiązania poruszającego temat symulacji w związku z tematyką prowadzonych zajęć projektowych na przedmiocie „Symulacja komputerowa”. Zespół przygotował rozwiązanie pozwalające na dokonywaniu reakcji chemicznych na podstawie odczynników chemicznych. Projekt posiada również implementację pozwalającą na poszerzenie wiedzy teoretycznej w zakresie chemii oraz tablicę układu okresowego pierwiastków. Jako jednostkę wykorzystujemy mole:

**Mol -** podstawowa w układzie SI jednostka liczności materii o symbolu (oznaczeniu) mol.

Liczba ta jest nazywana liczbą Avogadra.Definicja ta została zaaprobowana w listopadzie 2018 r., zmieniając starą definicję mola opartą na liczbie atomów w 12 gramach węgla 12C. Przed 20 maja 2019 roku, kiedy definicja ta weszła w życie, przyjmowano, że jeden mol jest to liczność materii układu zawierającego liczbę cząstek (np. atomów, cząsteczek, jonów, elektronów i innych indywiduów chemicznych) równą liczbie atomów zawartych w dokładnie 0,012 kilograma izotopu węgla 12C.

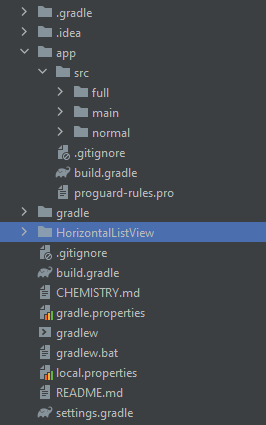
Rozwiązanie pozwala na wykonanie poniższych reakcji chemicznych:



# Przedstawienie projektu

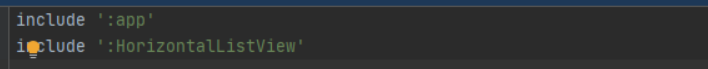
# Część praktyczna

Kod został podzielony na strukturę przedstawioną na poniższym rysunku:



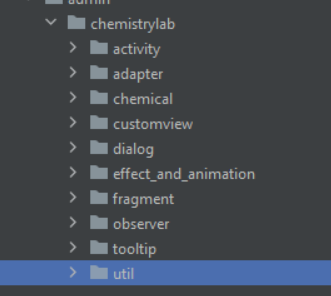
Jak możemy zauważyć kod jest podzielony na foldery widoczne na powyższym rysunku. Głównym sercem prezentowanego projektu jest folder app gdzie znajduje się główna część logiki przedstawiająca działanie poszczególnych jego elementów. Folder HorizontalListView odpowiada za widok listy prezentowany po lewej stronie, zawiera on pliki xml opisujące zachowanie w stosunku od rozdzielczości. Foldery z .gradle oraz gradle czy .idea odpowiadają za odpowiednie zależności importowane na potrzeby projektu. Plik local.properties ma na celu wskazanie ścieżki do SDK ponieważ korzystamy z Javy.

W pliku settings.gradle includujemy foldery app oraz HorizontalListView celem jego wykorzystania wraz z wcześniej wspominanym gradle.

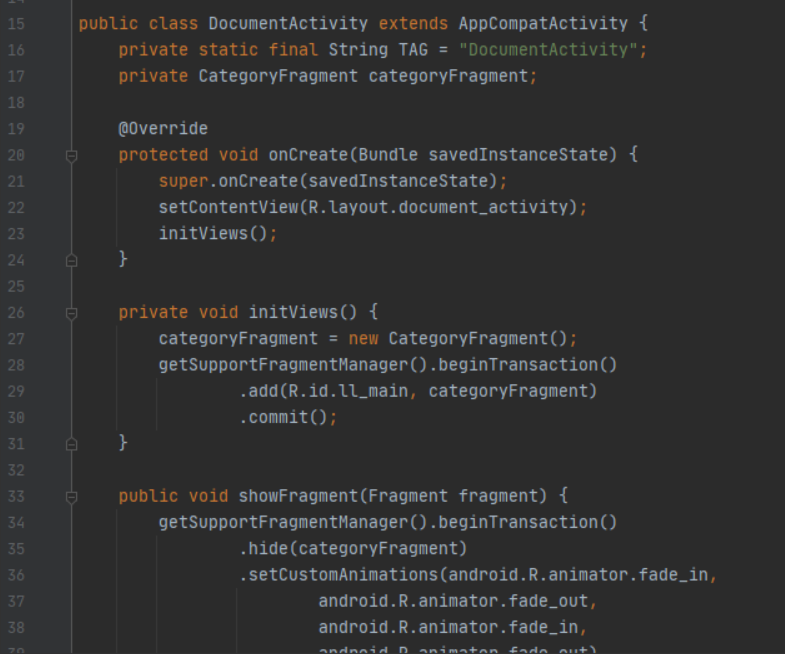


Kolejnym elementem wartym omówienia jest folder full gdzie występują trzy podfoldery: assets, java oraz res. Folderze assets przechowujemy pliki pdf.

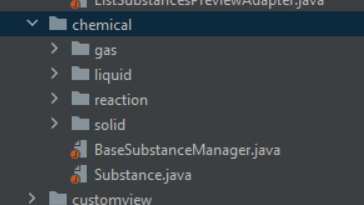
Folder java zawiera główną część prezentowanego projektu odpowiadającą za dokonywanie reakcji chemicznych.



Celem obsługi elementu odpowiedzialnego za wiki wykorzystano klasę DocumentActivity rozszerzającą AppCompatActivity. Poniżej prezentuje się kod tej klasy.



Folder chemical odpowiada za elementy prezentujące stan skupienia oraz reakcję pomiędzy nimi.



Wyróżnione tutaj są trzy stany skupienia gdzie każdy reprezentuje inny folder oraz folder odpowiadający za reakcję wspominanych czynników.

Folderem o którym warto również wspomnieć na przestrzeni pracy jest folder z utilsami, gdzie rozwiązane zostały problemy związane z przykładowo konwertowaniem symbolu czy pikseli.

# Podsumowanie

Prezentowany projekt jest rozwiązaniem z zakresu symulacji pozwalającym na poszerzenie wiedzy z zakresu wiedzy chemicznej. Pomimo problemów napotykanych na drodze niniejszego projektu został zaprezentowany oraz opracowano niniejszą dokumentację wraz z przybliżeniem tematyki.