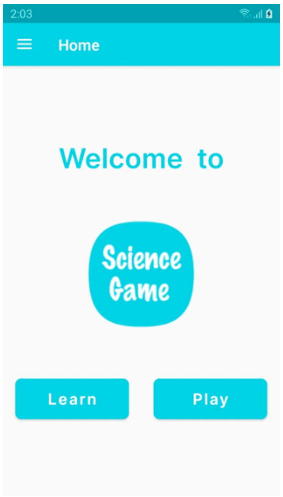
Report

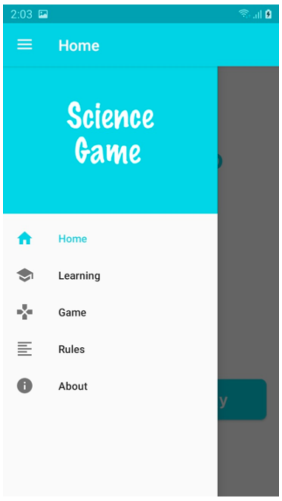
Outstanding presentation and discussion of design

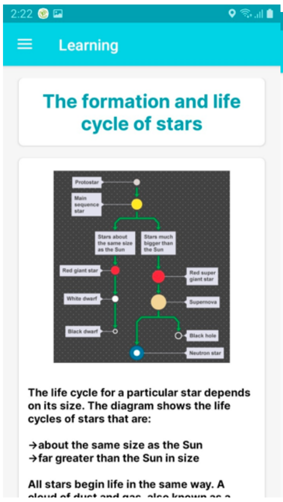
The application design is very simple that a children can use our application very easily. The animations are used to attract the users so that they play our game again and again. Light Blue color is used which helps for refreshing the eyes.
There is a Menu for navigating it to different screens and there are 5 screens for navigation.

- Home
- Learning
- Game
- Rules
- About

Here are the some of screen shots





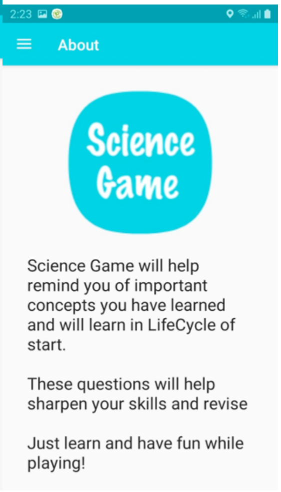


The formation and life cycle of stars

The life cycle for a particular star depends on its size. The diagram shows the life cycles of stars that are:

→about the same size as the Sun
→far greater than the Sun in size

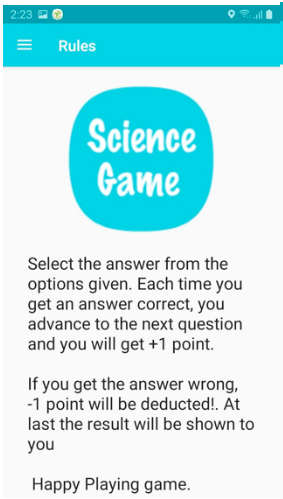
All stars begin life in the same way. A cloud of dust and gas, also known as a



Science Game will help remind you of important concepts you have learned and will learn in LifeCycle of start.

These questions will help sharpen your skills and revise

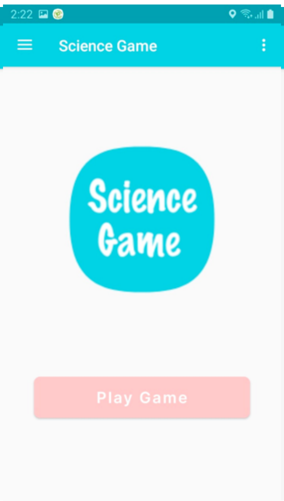
Just learn and have fun while playing!

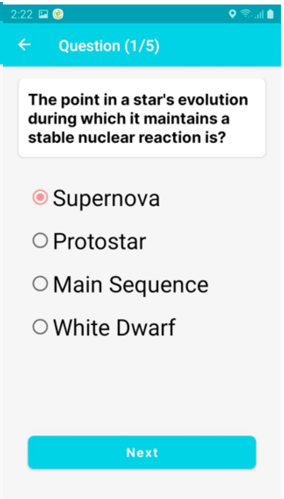


Select the answer from the options given. Each time you get an answer correct, you advance to the next question and you will get +1 point.

If you get the answer wrong, -1 point will be deducted!. At last the result will be shown to you

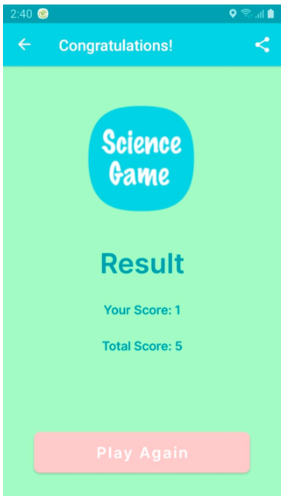
Happy Playing game.





The point in a star's evolution during which it maintains a stable nuclear reaction is?

● Supernova
○ Protostar
○ Main Sequence
○ White Dwarf



Your Score: 1

Total Score: 5

Very good discussion of OOP theory applied

The **Object-Oriented** Approach is used for storing the data of Quiz Question and Learning Details. I have used this class for storing the data and retrieving the data in the application. I have made a class **Sample Data** to store both There are two model class that are used, the one is **Guide** and the other is **Question.**

The **Guide class/Object** is used for text and picture resource that helps to understand the details of Lifecycle of Star.

```
data class Guide(
   val text: String,
   val imageResourceId: Int)
```

The **Question class/Object** is used to store/retrieve questions for MCQS.

```
data class Question(

   val text: String,

   val answers: List<String>)
```

Outstanding, excellent or very good understanding of the mobile development platform, Android API, the Kotlin language demonstrated.

As an Android Developer, kotlin is one of my favorite language for mobile application development because of its short structure than JAVA. I did not use any of the 3rd party libraries. Some of things that I like the most include

- Navigation Component Feature for navigating to different screen rather than intent
- Data Binding Feature for binding the data rather than find the view and then set text to it.
- Extension function for extending the functionality of a view rather than creating custom views.
- Variable defining feature is the one that I love most (var,val) rather than defining the datatype.
- Null Safety feature is very helpful for capturing the null values at compile time.

Very good critique of Android development with Kotlin with clear comparisons to other software development experiences.

Criteria	**Flutter**	**Kotlin**
Definition	Cross-platform app development framework from Google	Statically-typed, object-oriented programming language targeting JVM and JavaScript
Developed by	Google	JetBrains
Initial Release	2017	2011
Supported Platforms	Android, iOS, Google Fuchsia, Web, Linux, macOS, Windows	Android, JVM, iOS, macOS, Windows, Linux, JavaScript, WebAssembly
Learning Curve	Must learn Dart to build an app	More similar to Java, C#, Python, JavaScript – easy learning curve
App Size	Bigger	Less compared to Flutter
Performance	Native performance on all platforms	Slower compilation but overall unified experience across all platforms
Special features	Hot reload, fully customized widgets, and platform awareness for UIs	Interoperability with Java, concise, safe, and functional programming support
Special features	Hot reload, fully customized widgets, and platform awareness for UIs	Interoperability with Java, concise, safe, and functional programming support