Høyskolen Kristiania

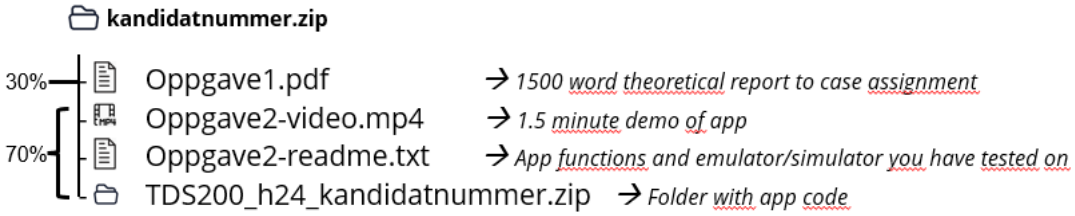
Course code: TDS200
Subject name: Kryssplattform
Exam format: Written home exam individually or in group up to 3 members
Delivery/submission date: 22.11.2024
File format: .zip

Remember to state your candidate number on your delivery folder.

Høyskolen Kristiania

EXAM
TDS200 Kryssplattform
Autumn 2024

You are required to complete two tasks, and your performance will be evaluated based on both technical and theoretical competence. The following pages contain the details of the two tasks. Your submission should follow the below structure:



Important Information

- This home exam can be solved and delivered individually or in group up to 3 members. It is expected that a group submission shall accomplish more, i.e., in terms of code volume and complexity etc than individual submission.
- Assessment guide for internal and external assessors, also as a reference for students, can be found at the end of this exam text.
- Read through the entire exam text before starting.
- Always follow good referencing practice. When using quotes, code snippets, statistics, images, figures, or similar elements that you have not created yourself, it is essential to provide proper citations. Follow the referencing guidelines established by Høyskolen Kristiania.
- Plagiarism must not occur. It is not allowed to copy code without providing a proper reference. Using TDS200 course lecture code is not regarded as plagiarism. However, you should also give citation when you refer to the lecture code.
- Using component library and npm modules is not considered plagiarism, and it is encouraged to demonstrate your competence utilizing npm modules and third-party libraries. Remember to provide appropriate references too.

Remember to state your candidate number on your delivery folder.

Høyskolen Kristiania

Task 1 – Theoretical competence (weighting: ~30%)

For this task, please read carefully through the following letter from CEO of OpenCanvas Studio, and write a report that consists approximately 1,500 words, excluding the reference list.

Subject: Kickoff of ArtVista App

Dear team,

We are excited to announce the launching of our latest project, ArtVista, an innovative virtual artwork hub and digital exhibition platform. ArtVista aims to transform the traditional gallery experience by providing a digital platform for artists worldwide to showcase their work, and for art enthusiasts to explore and engage with these artworks without the constraints of geographical and boundaries.

Our skilled development team, consisting of three full-stack developers proficient in JavaScript/TypeScript, will be responsible for this project. Your extensive experience in both web and mobile solutions is crucial as we build a solution that spans multiple platforms, ensuring a universally accessible and engaging user experience.

Through ArtVista, artists can upload their artwork to the digital platform, making them instantly available to a global audience. Art enthusiasts can browse the gallery extensively, learn about the artist's background, and discover related exhibitions. The platform also enables direct communications between artists and art enthusiasts, fostering a community-driven environment where users can discuss artworks, share insights, and participate in voting for featured artists and artworks.

Understanding our customers is important. We are set to launch a detailed market research to gather data on the distribution of Android and iOS users in Norway, serving as a basis for our prioritized platform. With the rapid growth in digital art platforms, time to the market is crucial. Additionally, this project must align with our resources and manage costs effectively.

Our goal is to develop an application that is not only visually appealing and user-friendly but also prioritizes high-quality content. Our development strategy will emphasize efficiency, scalability, and security, with a reliance on robust cloud solutions to support our needs.

Best Regards,

OpenCanvas Studio

Remember to state your candidate number on your delivery folder.

Høyskolen Kristiania

Given the information above and the requirement specification below, your assignment will be to formulate and present your idea as the “Tech Lead” regarding the most suitable approach for the development of ArtVista app.

Requirement specification of ArtVista (no priority order by numbering)

#	Function
1	The app must communicate with a backend service, e.g., Firebase.
2	Users should be able to view the digital artwork gallery in a list or grid format.
3	Users should be able to access the artwork details by clicking it.
4	Artists should have the opportunity to upload their artwork in a secure and intuitive way, and new artwork uploads may include images, descriptions, hash tags, artist profile, related exhibition.
5	New photo uploads may include images, descriptions, hash tags, locations, and date information.
6	Users can leave comments, share insights, and voting for featured artists and artworks.
7	Users shall be able to register and log in with email/password, ensuring that only authenticated users can access details and create new entries.
7	Seamless navigation between pages is expected.
8	The app should feature a user-friendly visual design and styling inspired by art world.
9	The app should be accessible on both Android and iOS platforms to cover the entire mobile market

Remember to state your candidate number on your delivery folder.

Høyskolen Kristiania

Task 2 – Technical competence (weighting: ~70%)

For this task, you are going to create an application called ArtVista.

Key evaluation criteria for the exam:

- The solution must be built and run without errors and can be run with Web browsers, as well as Android emulator and/or iOS simulator.
- Comment the code to illustrate your understanding of code and explain complex logic when necessary.
- The code should adhere to best practices to enhance code quality, including proper naming conventions, code structures and emphasizing code reusability.
- Ensure the app is user-friendly with all functionalities working as expected.
- Any external libraries, modules, or packages required for the project should be integrated appropriately.

Creation and delivery of exam project:

- Create your own Firebase and there must be some data available at Firebase so that the exam sensor can see that the solution works.
- Make a short video (screen recording without sound) of the app showing the functions you have implemented. Limit the video to no more than 1.5 minutes and ensure it highlights any advanced features that may affect the grade.
- Please also attach a file called Task2-readme.txt to describe what functionality you have implemented in the app and whether you have run it on Android emulator, iOS simulator or both.
- Remove the node_modules folder before uploading to WISEFLOW!
- Make sure the package.json file contains all npm dependencies, expecting the exam sensor to install npm packages with the command “npm install”
- The project folder you delivered shall be named as “TDS200_h24_your candidate number.zip”

Case: ArtVista App

This technical task is built on top of the theoretical analysis from Task 1. Regardless of your conclusion made for Task 1 about the suggested development approach, you should develop the **ArtVista** application using either **Ionic + Vue** or **React Native** for frontend development (in alignment with the technologies taught in class), and **Firebase** as Backend as a Service.

Remember to state your candidate number on your delivery folder.

Høyskolen Kristiania

Four primary requirements for implementation (no priority order by numbering):

- **Artwork Viewing**: Users can browse the digital gallery that displays an array of artworks, presented with a photo and abstracts in either list or grid format. Each artwork is clickable for a detailed view where users can learn more about the artwork.
- **Artist Uploads**: Artists can upload their artworks (including pictures directly from camera or from photo gallery, and additional information such as caption, description, etc), making them instantly available to a global audience.
- **Authentication**: Allow users to register and log in with email/password, ensuring that only authenticated users can access the details and create new entries.
- **Seamless Navigation**: Allow users to navigate seamlessly between different pages, i.e., the artwork overview, detailed view, upload artwork, and authentication.

The four points mentioned above present the minimum requirement for the assignment. For those aiming for a higher grade, more functionalities and complexity in the implementation is expected. Please check below list to see examples of functionalities that can contribute to a higher grade. Note that the app should be designed for the sensor to easily test with minimal manual configuration.

Examples of extended functionalities that can achieve a higher grade:

- **Enhanced Authentication**: The app supports multiple login options, including email/password and social logins such as Google Sign-In.
- **Access Control**: Implementing router guards that allows guest users to access the digital gallery overview, while only logged in users can access artwork details and upload new artwork, etc.
- **Interactive Communication**: Users can comment on the artwork, delete their own comments, provide like or dislikes, and participating in community voting for featured artists and artworks.
- **Search and Filtering**: Users can search and/or filter artworks based on keywords, such as category, artist, etc.
- **Artist Profiles**: Detailed profiles for each artist are accessible, linking directly to their associated artworks.
- **Related Exhibition:** Artworks may be linked to related exhibitions, which include geolocation data displayed on a Maps view for easy location viewing.
- **User Feedback**: Immediate feedback is provided during user interaction (error messages, loading indicators, etc.) to enhance the overall user experience.
- **User Accessibility:** The app features a user-friendly visual design and is usable by everyone, including those with disabilities. This includes careful attention to font sizes, color contrasts, and other design elements that enhance usability for all users.

Høyskolen Kristiania

Assessment Guide
TDS200 Kryssplattform
Autumn 2024

Task 1 – Theoretical competence

In Task 1, the candidate must assume the role of "Tech Lead" in a fictitious company that develops a digital art gallery application called "ArtVista". In around 1500 words, the candidate(s) is required to discuss and debate the most appropriate app development approach for this project, considering the provided requirement specification.

Expectations for the content and level of answer:

- The candidate(s) is expected to use combination of own experiences and external sources including online materials, research articles, blog posts, video content etc., to support the conclusion.
- The candidate(s) is expected to demonstrate a solid understanding of different app development approaches, highlighting their advantages and limitations.
- The candidate(s) is expected to incorporate relevant data and statistics for Android and iOS usage in Norway, as well as the adoption rate of cross-platform technologies., into the analysis.

Relevant working material may include (Posted on canvas page):

- Analyzing User Experience in Mobile Web, Native and Progressive Web Applications - A User and HCI Specialist Perspectives
- BridgeTaint: A Bi-Directional Dynamic Taint Tracking Method for JavaScript Bridges in Android Hybrid Applications
- End Users' Perception of Hybrid Mobile Apps in the Google Play Store
- A Survey and Taxonomy of Core Concepts and Research Challenges in Cross-Platform Mobile Development
- The Impact of Cross-Platform Development Approaches for Mobile Applications from the User's Perspective
- "React Native at Airbnb" by Gabriel Peal (2018) - https://medium.com/airbnb-engineering/react-native-at-airbnb-f95aa460be1c
- "Architecting Mobile Web Apps" by Michael Bleigh (Google I/O'19) - https://www.youtube.com/watch?v=NwY6jkohseg

Høyskolen Kristiania

Task 2 – Technical competence

For Task 2, the candidate(s) must implement an application called ArtVista using either **Ionic + Vue** or **React Native** for frontend development (in alignment with the technologies covered in the course), and **Firebase** as Backend as a Service. The candidate must also deliver a maximum 1.5-minute screen recording without sound showing the functions the candidate(s) has implemented.

Expectations for the content and level of answer:

- Four primary requirements for the implementation must be presented. These constitute the minimum requirement for the task. If only these four primary requirements are met, the candidate(s) can expect a maximum grade of C, given that delivered code fulfilled a very high standard. To achieve a higher grade above C, the student must demonstrate skills and knowledge beyond these four primary points.
- A list of examples for extended functionalities is attached to the assignment. An extremely good solution will include most of these advanced features, and/or functionalities of similar complexity and nature.
- Code snippets taken from external sources will not directly contribute to the grade. However, below two will contribute to the grade:
  - if the candidate(s) has made major changes to the code snippet and has adequately documented this comprehension.
  - The code originates from component libraries and is used as intended in the documentation.

Appendix: Grading scale

In the exam text, the approximate maximum score per task is shown as below:

Symbol	Fra eksamensavdelingen
A	90-100 poeng
B	80 – 89 poeng
C	60 - 79 poeng
D	50 - 59 poeng
E	40 - 49 poeng
F	0 - 39 poeng

- Task 1 = ~30 points
- Task 2 = ~70 points
- Total = 100 points

Remember to state your candidate number on your delivery folder.