Chirp! Project Report

ITU BDSA 2024 Group 6

Benjamin Ormstrup beor@itu.dk
Marcus Frandsen <megf@itu.dk
Victor de Roepstorff vicd@itu.dk
Valdemar Mulbjerg vamu@itu.dk
Christian Jörgensen chpj@itu.dk

1 Design and Architecture of Chirp!
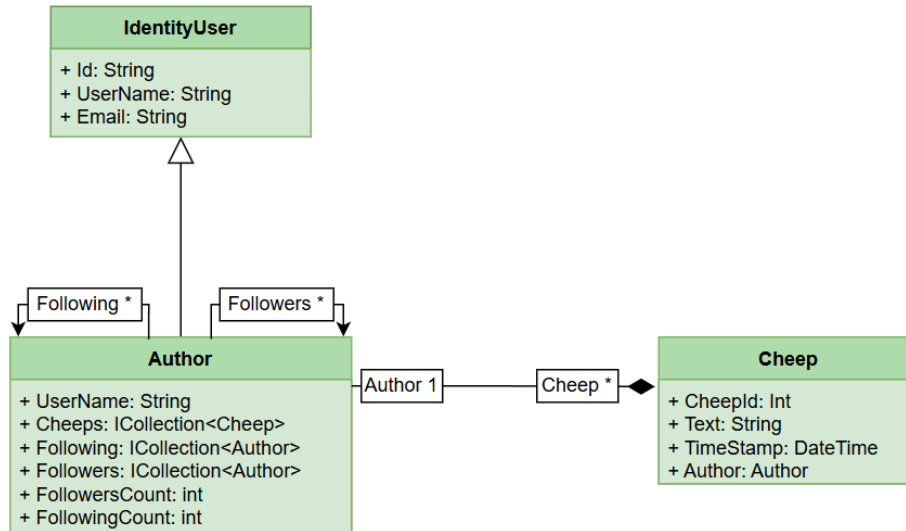
1.1 Domain model

Figure 1: *Illustration of the Chirp! domain model*

The UML class diagram above provides an overview of the core domain model for our *Chirp!* application, highlighting the primary entities, their attributes, and the relationships between them. A key aspect of the diagram is the cardinality between the entities, which defines how objects in one class relate to objects in another.
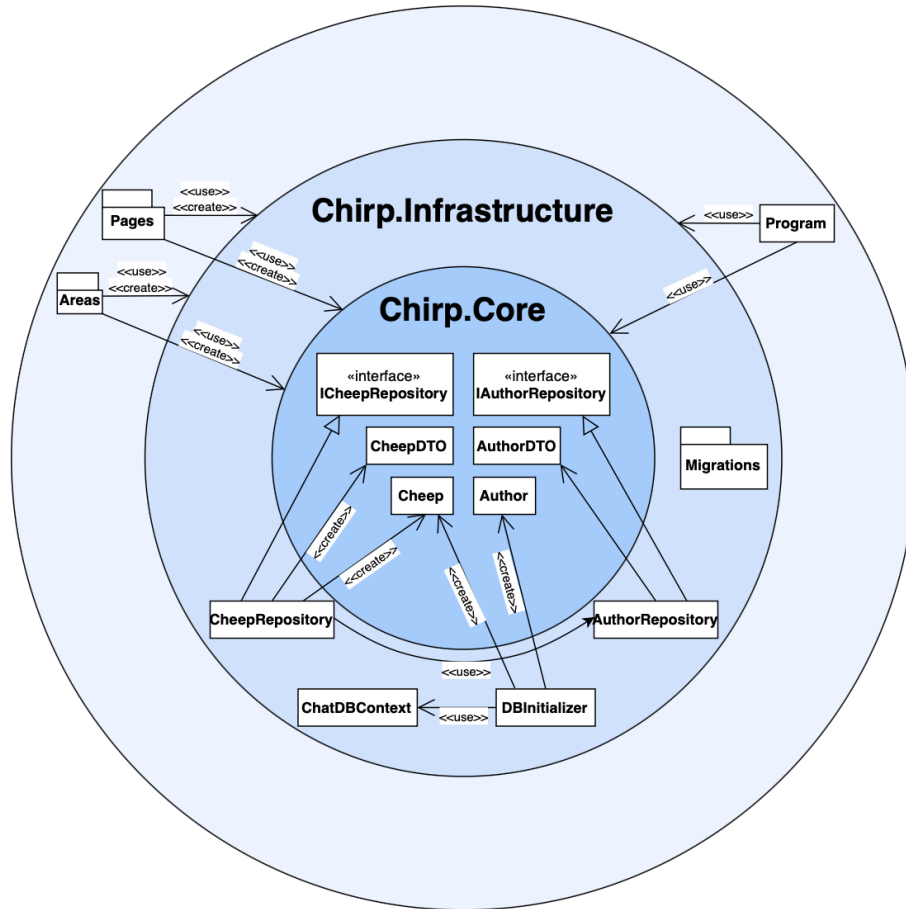
1.2 Architecture — In the small

Figure 2: *Illustration of the Chirp! architecture*

The union architecture diagram above visually represents the layered structure of the Chirp! application. The diagram consists of three circles with a different shade of blue, each symbolizing one of the core architectural layers: Core, Infrastructure, and Web. The components within each circle represent the key parts or responsibilities of that layer. The arrows throughout the diagram represent the unidirectional dependency flow of the application, where each layer depends only on the layers inside of itself:

- The Core layer is independent and does not depend on any outer layer.

- The Infrastructure layer relies on the Core while still remaining separate from the Web layer.

- The Web layer depends on both the Infrastructure and Core layers to deliver functionality to the user.

This layered structure ensures separation of concerns, making the program easily maintainable, testable, and scalable. Each layer can be adjusted, without having a direct impact on the logic and functionality of the layers above it.

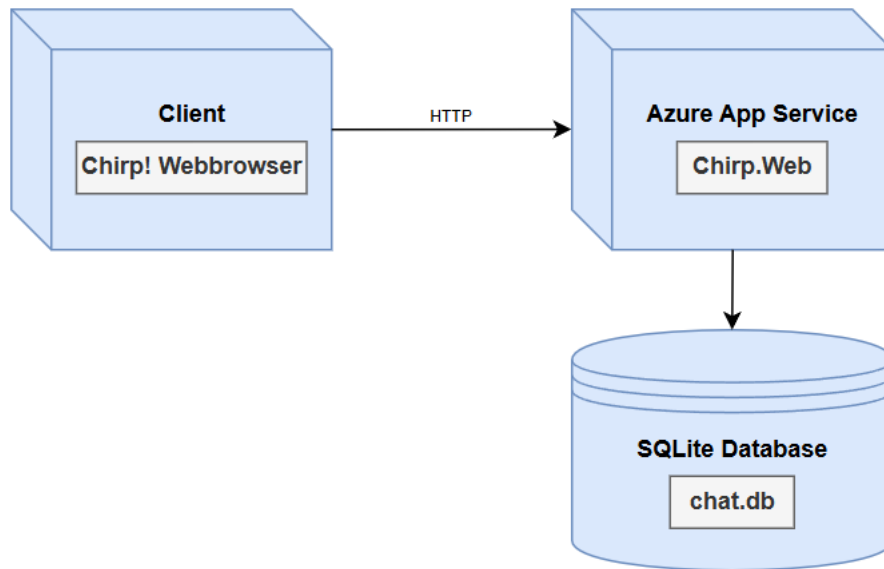## 1.3   Architecture of deployed application

Figure 3: *Application Architecture diagram of the deployed application*

Above is a diagram that illustrates the architecture of our deployed Chirp! application which focuses on the client-server relation.

1.4 User activities

The following two figures illustrate distinct user journeys through the Chirp!
application, tailored for both unauthorized users(not logged in) and authorized
users (logged in). Each figure maps out two specific journeys, showing how users
can interact with the system's key features.
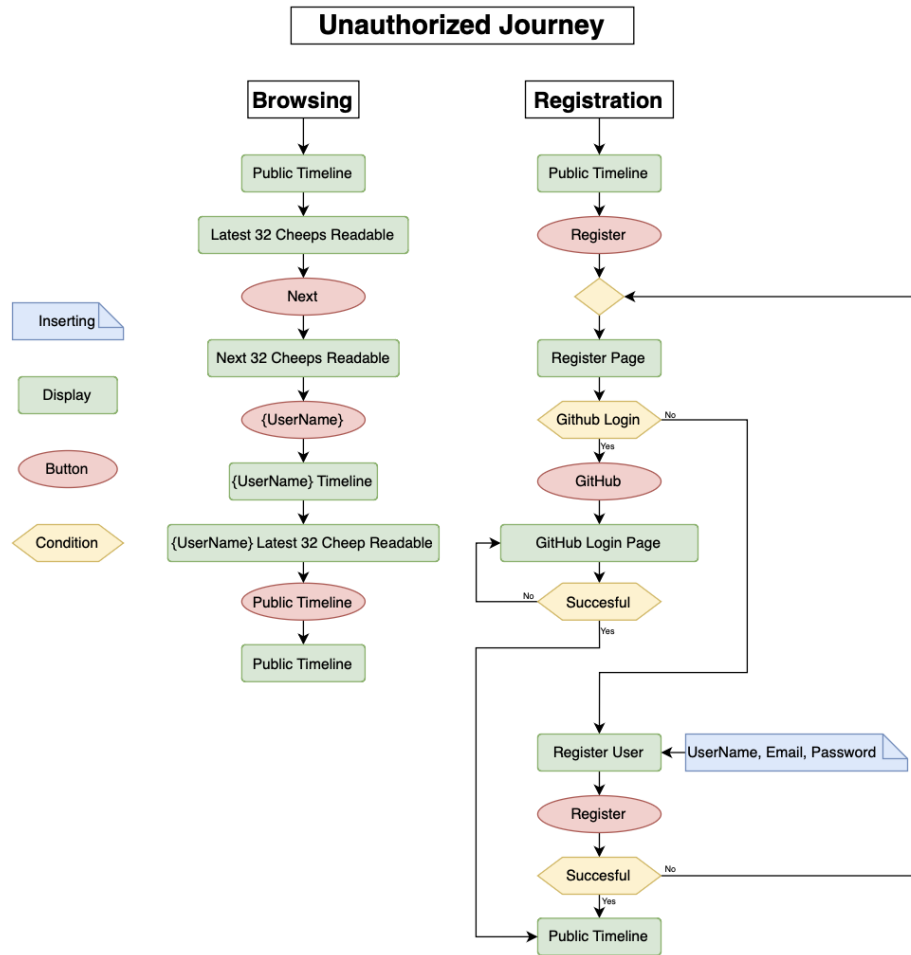
1.4.1 Unauthorized Journey

Figure 4: *Activity diagram of an unauthorized users journey*

This diagram focuses on the experience of users who are not logged into the
system.

The user journey on the left, "Browsing", highlights the program's accessibility

for unauthenticated users, allowing them to explore content without having to register first. The user can navigate the public timeline as well as other authenticated users timelines.

The user journey on the right, "Register", emphasizes the application's user-friendly onboarding process, using GitHub OAuth for convenience and ensuring new users can easily register as a user in the system. After registering the user is automatically logged in and navigated to the public timeline.
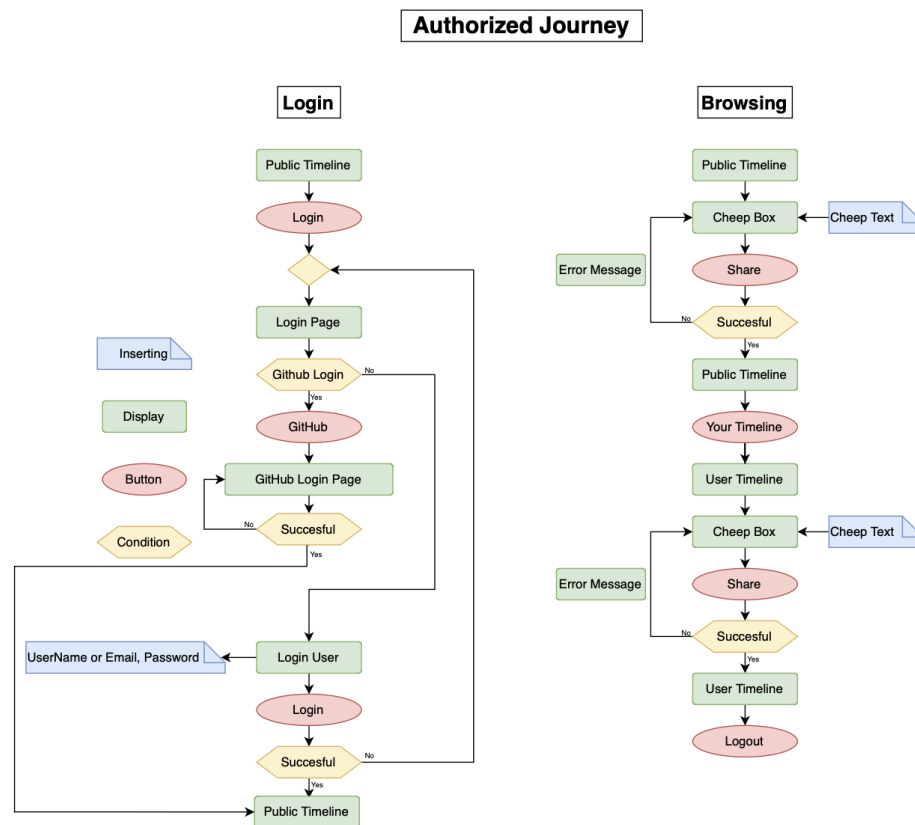
1.4.2 Authorized Journey

Figure 5: *Activity diagram of an authorized users journey*

In contrast to the previous figure, this diagram depicts the program's accessibility for a user who is already registered with the Chirp! application.

The user journey on the left, "Login", showcases the login process of an authenticated user, ensuring easy access to the full functionality of the program by allowing users to login with both username and email as well as a separate GitHub option.

The user Journey on the right, "Browsing", highlights the interactive features available to logged-in users, such as posting content and engaging with their own timeline, which form the core functionality of the Chirp! application.

2   Process

3   Ethics