

–B. Sc. Thesis Proposal*

Evaluation of requirements and implementation of a modern UI Editor

Freie Universität Berlin

Institute of Computer Science

Human-Centered Computing (HCC) Research Group

Matthias Kind

Student Number: 5338650

matthias.kind@fu-berlin.de

Date of Proposal Submission: 2.11.2022

Version: 2

Supervisor:

Florian Berger[†], Freie Universität Berlin, Germany

Examiner:

Prof. Dr. Claudia Müller-Birn[‡], Freie Universität Berlin, Germany

<Prof. Dr. Second Examiner[§]>, Freie Universität Berlin, Germany

*Template Version 2.0 — 2021-07-14

[†]Department of Mathematics and Computer Science, Human-Centered Computing Research Group

[‡]Department of Mathematics and Computer Science, Human-Centered Computing Research Group

[§]Department of Mathematics and Computer Science, <Name of Research Group>

Summary

The goal of the proposed thesis is to conceptualize and develop an UI Editor to create and configure Apps and Web-Apps for digital publishers. The Apps are based on an existing domain specific web framework, developed by <https://sprylab.com>. Apps are configured via dynamic resources, which contain all the styles, scripts and configs needed to render the customer's app on client devices. The concrete goal of the software is to enable the targeted user groups like internal developers, customer support and the customers (news and magazine publishers) to work more productive and confident.

In the context of HCI, the focus will be on applying different quantitative and qualitative methods to evaluate the user's needs and the state of the project, using agile development cycles to quickly test changes. Because this thesis will be written in a company with existing frameworks, customers and external APIs and flows that are fixed, all these methods get evaluated in a production environment with a lot of constraints, where the new Editor needs to be integrated into. While this limits the flexibility during the concept phase, it provides a large and diverse possible user base to see the effects and also perform quantitative surveys / Analytics.

Contents

1 Introduction 2

1.1 Motivation 2

1.2 Goal 3

2 Background 4

2.1 Context of the Project and Problem Description 4

2.2 Related Work 5

2.2.1 JSON Editor - generative UI 6

2.2.2 Sources for HCI methods and UI design 6

2.3 Research Questions 6

3 Methodology 7

3.1 Analyze 7

3.2 Design for Usability and early prototyping 7

3.3 Construct (Implementation) 7

3.4 Evaluation 8

4 Project Plan 9

4.1 Milestones 9

4.2 Timeline 10

5 Preliminary Outline 11

Bibliography 12

1 Introduction

The company SpryLab provides an software platform to publishers to provide their print- and digital content to their users. The user-facing part of that platform is an web framework based on Angular, which is rendered in Apps or as a Website and provides the customers components and data sources usually required by apps in this domain.

The app specific data is stored on "dynamic resources", which utilize a specific folder structure and contain common files used by web-apps, like static images, CSS and Javascript files, and the configuration files that declare the UI rendered by the app.

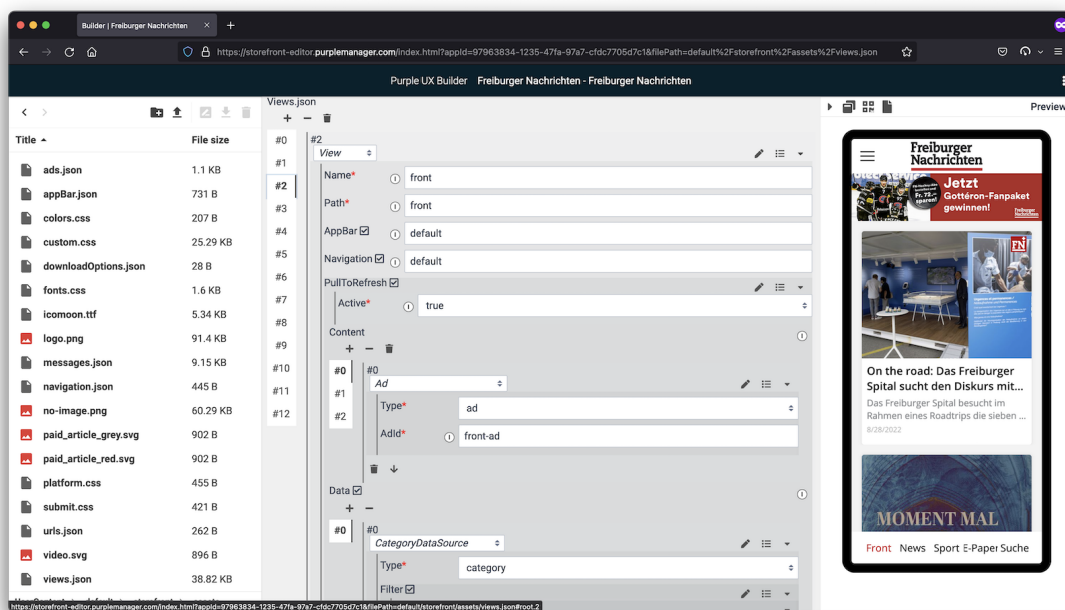
1.1 Motivation

Editing these dynamic resources by hand is tedious and error-prone, as the manual workflow consists of downloading a ZIP file, editing the contents and uploading without any validation before the resources are deployed to the app.

This requires a deep knowledge of the setup, what files and keys to put where, and still experienced users of the framework can easily introduce errors by misspelling a filename or putting a wrong component name inside the UI declaration files.

An existing attempt to have a web-based editor were not pursued with much ambition or proper requirements analysis to provide a pleasant user experience to users besides the original framework developers. Thus, current "non-power-users" often struggle with slow performance, missing explanations or cryptic error messages.

Besides the often unpleasant user experience, it also suffered from bad developer experience, like non-optimal project setups and the limits of existing libraries that were used, e.g. to edit the UI declarations based on specific schemata.



An example of the current editor UI

1.2 Goal

The goal is to give the different identified possible user groups an editor which enables them to work more productive, make less errors and get more interactive feedback from the system, so that there is less support needed by other entities like the framework developers.

This includes evaluating different HCI methods to evaluate the current state as well as the different needs of the users, and then using an agile development process to build a web based editor for the Purple Experience framework. The core of this will be an editor to edit the JSON files describing the App's UI, respecting JSON schema definitions and fitting the users different knowledge and skill levels.

On a more abstract level, the outcome of the thesis should give insights about integrating a new tool / UI into an existing production environment with many constraints, which methods and approaches worked and maybe also which failed.

The contributions I aim to produce with this bachelor thesis are:

software	an web app and backend that serves and present the editor to clients, possibly contributions to open source libraries if required to fulfill the needs of the editor.
HCI discoveries	documentation to the different methods and approaches used to gain the insight into the users, as well as evaluation of the results of these methods and how effective they proved in the context of changing a component inside a larger ecosystem.
user base knowledge	better knowledge about what the different user groups of the proposed editor are and can be, as well as their different habits, knowledge levels, common mistakes and more.

2 Background

This section gives some deeper background on the Web/App Framework, the editor I intend to build and what some common actions of users within the editor might be.

2.1 Context of the Project and Problem Description

In the following, I will only use the name "app framework" when referring to the meta-framework based on angular that can be deployed in Apps and as a Web-App.

To understand the use cases of the editor, I will first introduce you shortly to the app framework the editor should configure. As described in the summary, the framework's build output is loaded from so called "dynamic resources", together with the configuration files, styling in form of css files, static images and other web contents. At runtime, the framework reads the config files and renders the UI with a set of components. These components can be populated with data from datasources, change behaviour based on the "context", like device type, query parameters in the URL or other external factors.

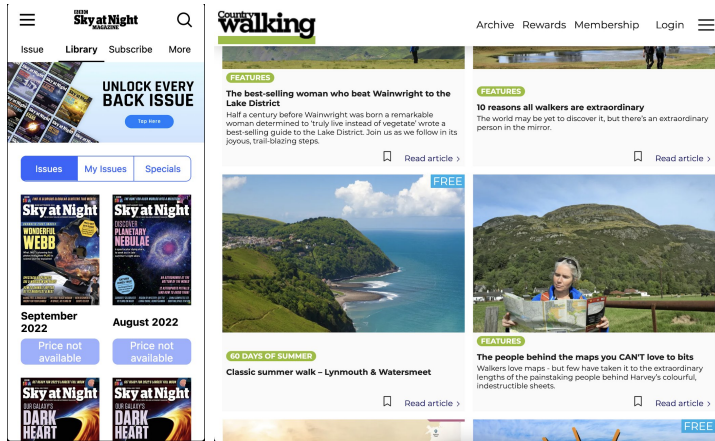
An simple example of an JSON configuration for a login view:

```
{
  "path": "login",
  "name": "login",
  "content": [
    {
      "type": "section",
      "class": "login-section",
      "content": [
        {
          "type": "html",
          "tag": "h1",
          "content": "LOGIN_TITLE"
        },
        {
          "type": "html",
          "tag": "p",
          "content": "LOGIN_IOS_TEXT",
          "condition": {
            "value": "$context.platform",
            "compareValue": "ios"
          }
        }
      ],
    },
    {
      "type": "button",
      "message": "Login",
      "buttonClass": "button",
      "tap": {
        "type": "login"
      }
    }
  ]
}
```

]
}

Even with this basic example without any data sources or other complex dependencies, it is quite hard to understand the mapping between configuration and generated UI, which leads to easily introduced errors.

Here are two screenshots of more complex UIs that can get configured with this framework:



The tasks performed depend a lot on the user; a admin from a publisher's company may just want to exchange an ad banner, rephrase some text or update a logo. The Customer Success team at sprylab does more complex configurations like adapting existng apps for new customers by exchanging messages, colors, sometimes even adding complete views or adapt sections inside. The framework developers use it for setting up new apps and features, confugiring complex filters for data sources and more.

Obviously, the easier the editor gets to use, the more the Customer Success and external users get enabled to do more changes on their own. But it needs to be noted, that this system is constrained by the complexity of the schemas defining the UI configs. As this editor needs to work with the existng schematas, there is a limit on how easy it will be to configure apps, but the goal is to push this boundary further than current workflows allow.

2.2 Related Work

This section consists of a literature review to situate your thesis in the scientific context. Which academic articles exist in your problem area, and how are they related to your work? When placing your thesis in the context of others, you need to consider other work, which uses a similar methodology or articles, who try to answer similar research questions.

The related work can be split into two (or even three) parts.

2.2.1 JSON Editor - generative UI

- Adamant: a JSON schema-based metadata editor for research data management workflows
- Understanding JSON Schema
- Interactive model driven graphical user interface generation
- JSON-GUI

Example Implementations which should get evaluated or taken as reference

- <https://github.com/json-editor/json-editor>
- <https://jsonforms.io/>

2.2.2 Sources for HCI methods and UI design

- Methods and Qualities of a Good User Interface Design
- Book: Lern human computer interaction, Christopher Reid Becker
- Book: Interaction Design: Beyond Human-Computer Interaction
- INTEGRATING HUMAN-COMPUTER INTERACTION DEVELOPMENT INTO THE SYSTEMS DEVELOPMENT LIFE CYCLE: A METHODOLOGY
- Brownfield Integration: Why It's Important For Modernizing Your Enterprise

2.3 Research Questions

How does an editor for dynamic resources for users with different levels of expertise look like and how can it be conceptualized and implemented within the constraints of an existing ecosystem?

Other questions that fit into the thesis topic:

- What pain points can be solved by existing libraries and tools, which require new development or enhancement?
- What challenges arise when applying HCI methods in a brownfield development process and how can they be dealt with?
- How can we improve the user experience for people with no knowledge about the system, while also giving powerusers the amount of flexibility they need?

3 Methodology

Ref: "Learn Human computer interaction", 104 diagram

3.1 Analyze

1st phase "discovery": assess current situation through various User Research methods (taken from "Learn Human computer interaction", 132: Human Centered Methods for User Research)

- fly-on-the-wall method: observe without users knowing they are observed
- moderated observation: Create scenario for user and note the way(s) the users do the task
- user interviews: prepare questions on workflows, what they are missing, what takes most time
- Quantitative survey

3.2 Design for Usability and early prototyping

Build deployable prototype using agile development methods

- Evaluate methods from 1st phase on effectiveness and continue using them with test circle of persons
- Use SCRUM to plan work
- use CI/CD to allow fast iterations after changed requirements
- Build in Analytics / Tracking service for automatic user data evaluation
- A/B testing?

3.3 Construct (Implementation)

This section is closely coupled to 3.2, this construction of the software results from the prototypes and is done in the same SCRUM sprints as the design.

The tech stack is one of the constraints imposed by the environment, the backend will be written in Javascript for NodeJS express as the web framework.

The frontend, including the JSON editor should be written in react to utilize existing knowledge from other developers at the company and to guarantee long support. The backend will be hosted as a docker container on a kubernetes cluster and easily deployable via gitlab Continuous integration systems.

Depending on the collected requirements during the first phase, we might also integrate REDIS as a light messaging bus to support multiple instances.

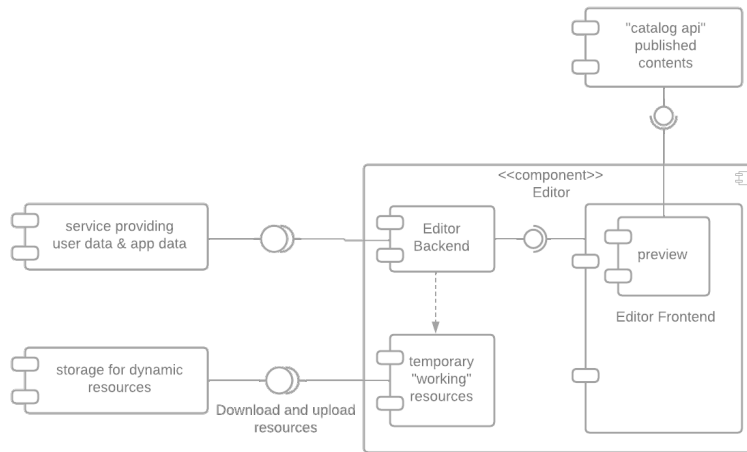


Figure 1: UML component diagram of the integration into the existing environment

3.4 Evaluation

The evaluation towards the end of the thesis consists of using some of the methods mentioned above a last time. There the focus will probably be on a survey and evaluating analytics data, as well as getting verbal feedback from test users. The results then get compared to the results raised earlier to identify improvements in the productivity of the users.

From that results, more abstract realizations about developing new software with HCI methods in constrained existing environments can be elaborated.

4 Project Plan

4.1 Milestones

Specify the milestones of your upcoming project. Please describe when you plan to achieve which milestone and what artifact(s) or outcome will result from each milestone. Also, keep in mind what the goal of each milestone is.

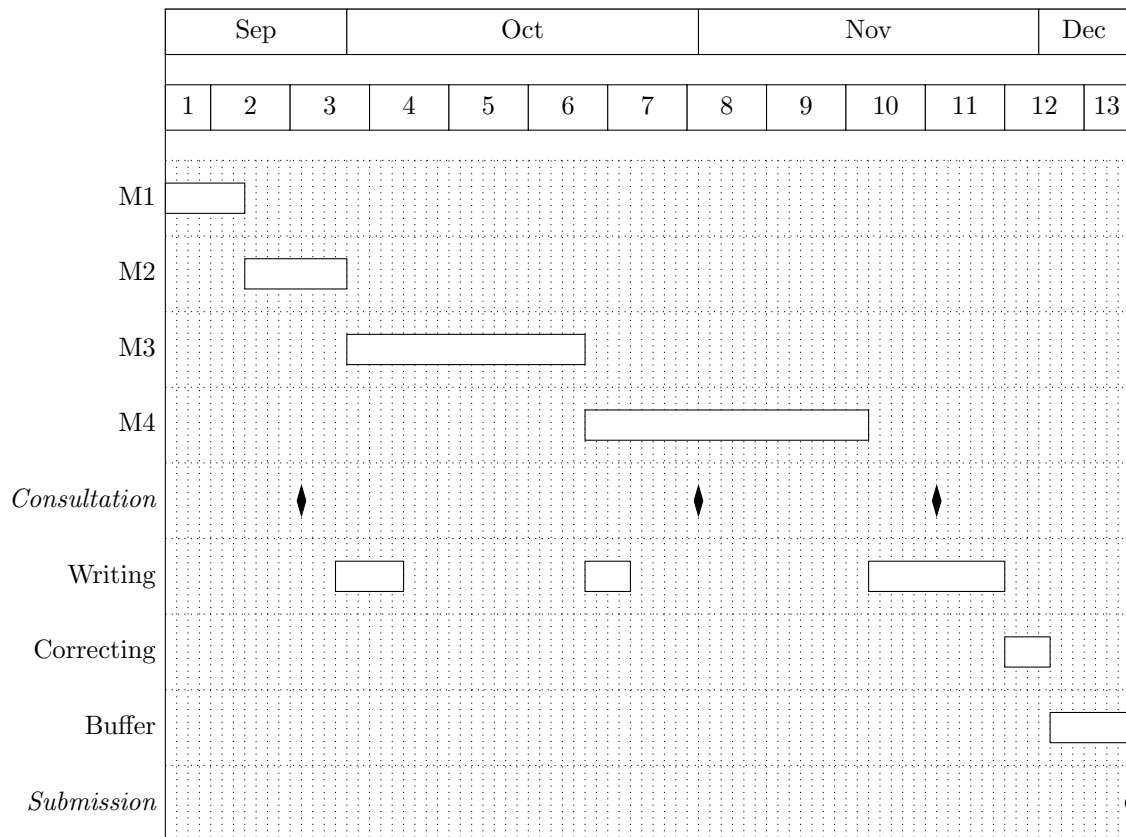
M1	Milestone — Literature and Source Review
Due date	2022-09-21 (Week 1)
Tasks	Identifying and read other studies/thesis/papers about UI editors, HCI methods, bronfield development Look through old HCI lectures
Outcome	A list of relevant papers, articles and possilby open source repositories A list of applicable HCI methods and when and with whom they should be applied A final text summarizing the main findings and approaches, which might be useful for my project.
Goal	General understanding of methods to build UIs and software with HCI methods in an constrained environment. Having a good foundation for discussing my results in the context of other people's work.

M2	First round of methods & first deployment of Low-fidelity wireframe prototype
Due date	2022-09-30 (Week 3)
Tasks	Conduct HCI methods to evaulate current state and initial requirements with at least 4 persons from at least 2 diffrent resorts. Prepare the codebases for front- and backend including unit tests, some basic UI wireframes to test layouts and diffrent pages
Outcome	A document condensing the outcomes of the survey / observations, including a TODO list derived from their initial requirements and the requirements of the company's software environment. A gitlab repository that can build a docker image ready to deploy on kubernetes.
Goal	Having laid the base work for the iterative work on the editor, as well as collected data for the evaluation at the end of the project.

M3	Deploy editor prototype with full views.json schema support
Due date	2022-10-21 (Week 5)
Tasks	Modify or build an Editor that reads the provided JSON schema for the UI configs and allows modifying the JSON files describing the UI. Adapt schema generators to support more metadata / annotations inside the code
Outcome	A editor view where users can manipulate the UI layout for an app. Performance should be better than exisitng JSON editor implementations with the usual schema and config sizes (validate using production app clones).
Goal	Having a solid foundation of one of the core features of the editor, to change the UI structure, while beeing more usable than exisitng solutions

M4	final deployment and final round of evaluation
Due date	2022-11-15 (Week 8)
Tasks	Conduct HCI methods to evaluate current state and initial requirements with at least 4 persons from at least 2 different resorts. Prepare the codebases for front- and backend including unit tests, some basic UI wireframes to test layouts and different pages
Outcome	A document condensing the outcomes of the survey / observations, including a TODO list derived from their initial requirements and the requirements of the company's software environment. A gitlab repository that can build a docker image ready to deploy on kubernetes.
Goal	Having laid the base work for the iterative work on the editor, as well as collected data for the evaluation at the end of the project.

4.2 Timeline



5 Preliminary Outline

1 Introduction

- 1.1 Motivation
- 1.2 Research goal and question
- 1.3 Research approach and methodology

2 Theoretical Background

- 2.1 Concepts and definitions of the Purple apps and the web framework
- 2.2 Introduction to the used HCI User Research methods
- 2.3 Introduction to the development process

3 Related Work

- 3.1 Related software
- 3.2 Related studies in this field

4 Analysis

- 4.1 Defining and conducting initial User Research
- 4.2 Collecting the structural requirements implied by the ecosystem
- 4.3 Derive requirements

5 Design Process

- 5.1 high- and low-fidelity prototyping

6 Implementation

- 6.1 Overview of system architecture
- 6.2 Technical implementation
- 6.3 Implementation of generative UI editor respecting JSON schemata
- 6.4 Using iterative development process methods in practice

7 Evaluation

- 7.1 present final conducted User Research methods
- 7.2 Present study results in comparison to initial state of the editing process
- 7.3 Evaluate study results regarding the thesis research question

8 Discussion

9 Conclusion

- 9.1 Limitations
- 9.2 Future Work

References