

Design Document

November 17 2016

Abstract

The Forge VR Explorer branches from an Autodesk prototype project called Vrok-It, which is a simple web-based 3D model viewer and mobile virtual reality (VR) explorer. The project will expand upon its ability to display uploaded 3D models in browser and in VR, and improve its accessibility. Conventionally, viewing 3D models in VR is a challenge if you have model files on many devices, or have a headset that only works in conjunction with a smart-phone. The Forge VR Explorer aims to do this by utilizing a web-based software that uses the features of the Autodesk Forge API. The project will also be expanded with new ideas and stretch goals as the project is developed.

TABLE OF CONTENTS

1	Overview	4
1.1	Scope	4
1.2	Purpose	4
1.3	Intended audience	4
1.4	Conformance	4
2	Definitions	4
3	Conceptual model for software design descriptions	5
3.1	Software design in context	5
3.2	Software design description within the life cycle	5
4	Design description information content	5
4.1	Introduction	5
4.2	SDD identification	5
4.3	Design stakeholders and their concerns	6
4.4	Design views	6
4.5	Design viewpoints	6
4.6	Design elements	6
4.7	Design overlays	7
4.8	Design rationale	7
4.9	Design languages	7

5	Design viewpoints	7
5.1	Introduction	7
5.2	Context viewpoint	7
5.3	composition viewpoint	8
5.4	Logical viewpoints	8
5.5	Dependency viewpoint	8
5.6	Information viewpoint	9
5.7	Patterns use viewpoint	9
5.8	Interface viewpoint	9
5.9	Structure viewpoint	9
5.10	Interaction viewpoint	10
5.11	State dynamics viewpoint	10
5.12	Algorithm viewpoint	10
5.13	Resource viewpoint	10

1 OVERVIEW

Tentative, unsure if this is needed: The software implemented is a cross-platform application capable of accomplishing several tasks. The software allows for the upload of CAD files, renders them on the website using the Forge API, allows the transfer and rendering of the models onto a user's mobile device, and then has the capability to allow the user to view the models in 3D with Google Cardboard. This document aims to delve into the design of each piece of functionality and seeks to expand its design and structure.

1.1 Scope

The scope of this document only covers the information regarding user experience data flows, the software design description and the design structure of the software. This document does not cover implementation decisions or specific quality requirements.

1.2 Purpose

The purpose of this software design document is to describe the user experience flows and provide the software design description to its intended audience. Additionally, this document provides a design framework that the developers will be using in order to assess the progression of the software through its development lifespan.

1.3 Intended audience

The intended audience of this software design document are the developers planning and building the software, and the stakeholders who include: Autodesk's Forge team, the clients and the advisors to the developers.

1.4 Conformance

2 DEFINITIONS

CAD Computer Aided design is software used to design and view 3D models.

CAD file The type of files that can be uploaded to the website for viewing in the Forge viewer. We will narrow down what types of files can be used as we progress through the development process.

FORGE [?] A collection of API services provided by Autodesk that provide 3D modeling services and tools.

Forge Viewer This is one of the APIs in Forge. It displays 3D models from CAD files and also allows for user interaction.

Model Derivative API This is another API in Forge. It can generate SVF files that we can utilize in the application from other model filetypes.

VR Acronym for virtual reality, typically a peripheral device or smartphone

exploding model A model viewing functionality that separates components from their original locations in order to gain an alternate view of the model.

3 CONCEPTUAL MODEL FOR SOFTWARE DESIGN DESCRIPTIONS

Not exactly sure what most of this section wants

3.1 Software design in context

3.2 Software design description within the life cycle

3.2.1 Influences on SDD preparation

The SRS is the main influence on the SDD as it holds the requirements and features determined necessary by all the stakeholders. This drives the design to meet those requirements to properly satisfy all stakeholders. However, this also means that it sets the design constraints.

3.2.2 Influences on software life cycle products

3.2.3 Design verification and design role validation

In order to verify that our software meets requirements, test cases will be used in order to walkthrough the system and demonstrate the actual functionality of the software system. In terms of user experience the verification and validation will be mainly done through some primary tests with Autodesk with final okay coming from the clients.

4 DESIGN DESCRIPTION INFORMATION CONTENT

4.1 Introduction

4.2 SDD identification

4.3 Design stakeholders and their concerns

The primary stakeholder for this project is Patti Vrobel who works for Autodesk. User experience is a primary focus of both Patti and Autodesk. The project addresses this by adding and altering features to make the website more accessible for users.

4.4 Design views

UML use case diagram is most likely what we will be using, we don't necessarily show the diagram itself here. But we do explain we are using it and how we are focusing on design details from a user perspective? Probably need Nels to clarify

4.5 Design viewpoints

- viewpoint name
- design concerns that are the topic of the viewpoint
- design elements, defined by that viewpoint, entities, attributes, etc
- analytical methods or other operations
- viewpoint source

4.6 Design elements

it can be any of the following: design entity, relationship, attribute or constraint. We'll have a giant list here I believe. It'll be all of the pieces of the system

4.6.1 Design entities

list of all the entities, their type, and purpose

4.6.2 Design attributes

provides a statement of fact about a design element (entity, relationship, constraint)

4.6.2.1 Name attribute: name of the element

4.6.2.2 Type attribute: description of the kind of element

4.6.2.3 Purpose attribute: Why it exists

4.6.2.4 Author attribute: identification of designer, basically kinda helps explain who is responsible for what elements

4.6.3 Design relationships

statement of fact about the association or correspondence between 2 or more entities. Each relationship will have name and type

4.6.4 Design constraints

rule or restriction imposed by one element (source) onto another element (target). Each should have a name and type

4.7 Design overlays

gives additional info about a design view

4.8 Design rationale

arguments and justifications about design decisions. Not to be confused hopefully with my scope section where I say we won't be going over implementation justifications

4.9 Design languages

5 DESIGN VIEWPOINTS

5.1 Introduction

the table in the document basically tells us what the concern with each section is as well as which UML diagram we'll need for each section, I'll go ahead and list the diagrams we'll probably use in here for us though

5.2 Context viewpoint

is meant to be a black box perspective, UML use case diagram will probably suffice here

5.2.1 Design concerns

5.2.2 Design elements

5.2.3 Example languages

5.3 composition viewpoint

UML package diagram, UML component diagram, UML deployment diagram

5.3.1 Design concerns

5.3.2 Design elements

5.3.2.1 Function attributes:

5.3.2.2 Subordinates attributes:

5.3.3 Example languages

5.4 Logical viewpoints

UML class diagram, UML object diagram

5.4.1 Design concerns

5.4.2 Design elements

5.4.3 Example languages

5.5 Dependency viewpoint

UML package diagram and component diagram

5.5.1 Design concerns

5.5.2 Design elements

5.5.2.1 Dependencies attribute:

5.5.3 Example languages

5.6 Information viewpoint

entity-relation diagram, UML class diagram

5.6.1 Design concerns

5.6.2 Design elements

5.6.2.1 Data attribute:

5.6.3 Example languages

5.7 Patterns use viewpoint

UML composite structure diagram

5.7.1 Design concerns

5.7.2 Design elements

5.7.3 Example languages

5.8 Interface viewpoint

UML component diagram

5.8.1 Design concerns

5.8.2 Design elements

5.8.2.1 Interface attribute:

5.8.3 Example languages

5.9 Structure viewpoint

UML structure diagram, class diagram

5.9.1 Design concerns

5.9.2 Design elements

5.9.3 Example languages

5.10 Interaction viewpoint

UML sequence diagram, UML communication diagram. This will be a HUGE section for us

5.10.1 Design concerns

5.10.2 Design elements

5.10.3 Examples

5.11 State dynamics viewpoint

UML stat machine diagram. This may also be a large section

5.11.1 Design concerns

5.11.2 Design elements

5.11.3 Example languages

5.12 Algorithm viewpoint

Don't think we will need this section at all

5.12.1 Design concerns

5.12.2 Design elements

5.12.3 Processing attribute

5.12.4 Examples

5.13 Resource viewpoint

UML real-time profile, UML class diagram, UML object constraint language

5.13.1 Design concerns

5.13.2 Design elements

5.13.2.1 Resources attributes:

5.13.3 Examples