

Slovenská technická univerzita
Fakulta informatiky a informačných technológií

Tímový projekt
ASICDE

Dokumentácia inžinierskeho diela

Predmet: Tímový projekt I.

Členovia tímu:
Bc. Dominik Dancs
Bc. Tadeáš Drahovský
Bc. Adam Chmara
Bc. Gergely Lengyel
Bc. Kamil Lihan
Bc. Lukáš Mišaga
Bc. Karolína Trnovcová

Vedúci tímu: Ing. Lukáš Kohútka, PhD.

Akademický rok: 2020/21

Obsah

Obsah	2
Úvod	3
Ciele pre zimný semester	4
Špecifikácia požiadaviek na celkový systém	5
Funkčné požiadavky	5
Nefunkčné požiadavky	5
Architektúra	7
Vývojové prostredie pre všetky súčasti projektu	7
Backend	9
Použité technológie	9
Moduly backendu	10
core	10
parser	10
commons-testing	11
commons-error-handling	11
commons-persistence	11
commons-security	11
commons-web	11
Aktuálny dátový model	12
repo	13
user	14
user_role	15
organization	16
organization_invite	17
team	18
team_membership	19
Frontend	20
Použité technológie	20
Homepage	21
Prehľad repozitárov	21
Dashboard organizácie	22
Testovanie	23
Prílohy	
Príloha A: API dokumentácia	
Príloha B: Frontend Mockup	
Príloha C: Frontend screenshoty	

Úvod

Vývojári číslicových systémov zväčša tieto systémy navrhujú v rámci jednoduchých textových editorov ako je napríklad Notepad++. Tieto editory však častokrát nie sú dostatočne robustné a vybavené na to, aby pokryli všetku požadovanú funkcionality, ktorá je vhodná alebo žiadúca pre týchto vývojárov. Editor, ktorý by predstavoval komplexnejšie a sofistikovanejšie vývojové prostredie by mohol byť riešením pre tento problém.

ASICDE je online editor, ktorý podporuje syntaktické zvýrazňovanie textu, automatické dopĺňanie textu, kontrolu syntaktických chýb a vývoj číslicového dizajnu integrovaných čipov, konkrétnie pre programovací jazyk SystemVerilog. Toto vývojové prostredie by okrem editovania a vizualizácie kódu umožňovalo aj kolaboráciu a komunikáciu v rámci skupiny ľudí alebo organizácie a jej tímov. Vďaka týmto funkcionalitym by vývoj rozsiahlejších číslicových systémov mal byť jednoduchší, rýchlejší a efektívnejší. Systém bude taktiež poskytovať možnosti predaja a kúpy jednotlivých modulov, ktoré používateľia alebo organizácie sprístupnia k predaju v rámci obchodu s licenciami na IP jadrá.

Tento projekt sme prevzali po predošlých študentoch, ktorí sa tomuto projektu venovali v rámci diplomových prác. Naimplementovali v ňom základné funkcionality editora, ako aj základné funkcie správy používateľov na backende. Kvôli tomu, že dané funkcionality neboli kompletne implementované a odladené, niekedy nespĺňali zadefinované konvencie a samozrejme, že vždy je možné systém zlepšovať, sme sa rozhodli v tomto projekte pokračovať. Našimi úlohami je ďalej vyvíjať tento systém, opraviť v ňom existujúce chyby a dokončiť neúplné funkcionality, vylepšovať ho, zvýšiť jeho bezpečnosť a pridávať do neho aj nové funkcionality.

Našou hlavnou motiváciou pre riešenie tohto projektu je práve jeho komplexnosť, zaujímavosť a vysoký potenciál byť užitočným pre mnohé firmy ako aj jednotlivcov. Ďalším dôvodom bola nedostupnosť iných dostatočne robustných nástrojov, ktoré na rozdiel od tohto systému boli často krát obmedzené, neintuitívne, ba dokonca nepoužiteľné v reálnom nasadení.

Tento dokument obsahuje informácie o systéme vytvorenom v rámci témy “Webové IDE pre ASIC [ASICDE]”. Zdokumentované sú v ňom aspekty ohľadom celkovej architektúry, ako aj samostatne časti backendu a frontendu, jednotlivé moduly a komponenty z ktorých sa skladá.

Ciele pre zimný semester

Pre zimný semester sme si stanovili nasledovné ciele:

- Rozbehanie a spustenie už existujúceho projektu
- implementovať automatické buildenie a testovanie
- kompletný refactoring kódu
- upravenie konfigurácií
- opravenie základných častí backendu - databáza, API
- navrhnutie mockupov obrazoviek pre frontend
- upravenie a implementovanie prvých obrazoviek (home page, login, registration)
- spísanie dokumentácií k projektu

Celkový pohľad po zimnom semestri

Za trvania zimného semestra sa nám podarilo vytvoriť upravenú formu existujúceho systému tak, aby splíňal všetky stanovené ciele, ktoré boli naň určené za dané obdobie.

Špecifikácia požiadaviek na celkový systém

Funkčné požiadavky

- Používateľ musí byť prihlásený aby mohol používať aplikáciu.
- Po registrácii, pred prvým prihlásením, používateľ musí potvrdiť svoju emailovú adresu pomocou linku ktorý dostane mailom.
- Po prvom prihlásení sa zobrazí tutoriál ktorý ukáže funkcionality aplikácie a možnosti používateľa.
- Používateľ môže byť členom iba jednej organizácie, alebo môže vlastniť organizáciu.
- Vlastník organizácie môže pozvať členov do svojej organizácie podľa e-mailovej adresy alebo používateľského mena.
- Po odoslaní pozvánky registrovaní používateľ je notifikovaný aj v rámci aplikácie, aj mailom. Používateľ, ktorý ešte nemá účet je notifikovaný mailom aby sa zaregistroval.
- V organizácii sú nasledujúce roly:
 - a. Owner - vlastník organizácie (vždy iba jeden) má plné právomoci
 - b. Admin - má možnosť vytvárať a spravovať tímy, nemá oprávnenia na modifikáciu samotnej organizácie
 - c. Member - člen organizácie má oprávnenie vidieť jej obsah a môže mať prístup aj do niektorých tímov alebo repozitárov
- V rámci organizácií sa dajú vytvoriť tímy, do ktorých sa dá pridať používateľ iba taký, ktorý je členom organizácie.
- V rámci tímu je možné vytvoriť repozitáre, na ktorých môžu pracovať iba členovia tímu.
- V rámci tímov tiež existujú role:
 - a. Team Admin - plné právomoci správy tímu (premenovanie, pridávanie členov, vytváranie a správa repozitárov)
 - b. Team Member - oprávnenie vidieť obsah tímu a modifikovať kód v repozitároch
 - c. Viewer - oprávnenie vidieť obsah tímu bez možnosti kontribúcie
- Každý používateľ má možnosť upravovať svoje osobné údaje a nastavenia účtu, a má možnosť svoj účet zmazať
- Každý používateľ má plné právomoci na svoje vlastné repozitáre, ktoré nespadajú pod žiadnu organizáciu
- Používateľ, teda vlastník repozitáru, má možnosť previesť vlastníctvo repozitára na iného používateľa

Nefunkčné požiadavky

- Odozva aplikácie musí byť menej ako 1 sekunda
- Riešenie musí byť nezávislé od operačného systému
- Systém musí byť možné použiť viacerými používateľmi naraz
- Systém musí byť jednoducho škálovateľný
- Systém musí byť responzívny
- Systém musí byť modulárny
- Systém musí spĺňať prísné podmienky dostupnosti
- Systém musí byť jednoducho konfigurovateľný

- API rozhranie spĺňa konvencie a štandardy REST
- Systém musí byť zabezpečený voči neautorizovaným operáciám
- API rozhranie neumožňuje získať citlivých údajov určených pre interné spracovanie (heslá a iné metadáta)

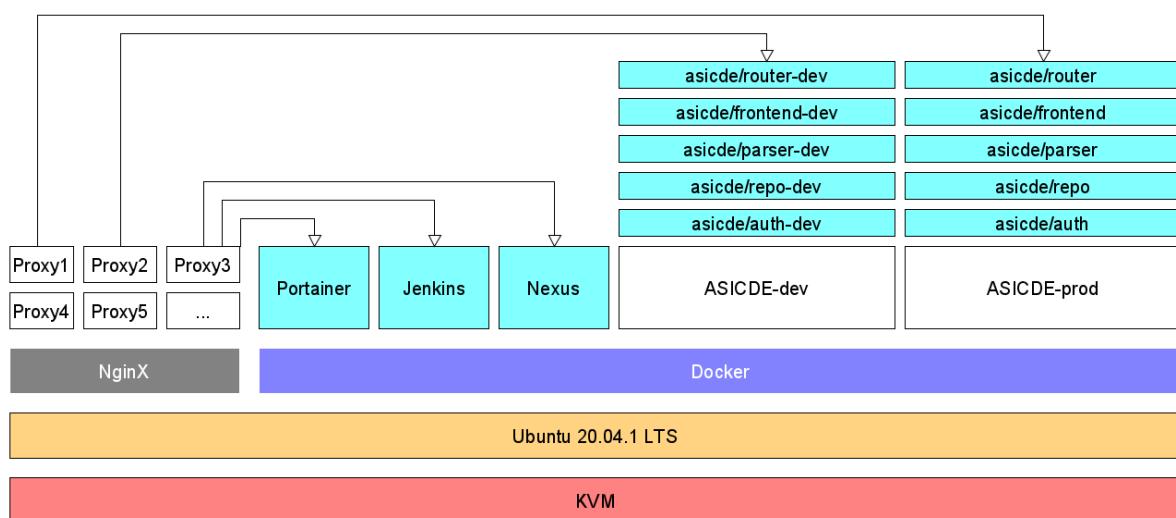
Architektúra

Našu aplikáciu vieme rozdeliť na dve hlavné časti:

- Backend - ktorá je implementovaná vo frameworku Spring (programovací jazyk Java)
- Frontend - je implementovaná vo frameworku Angular (programovací jazyk JavaScript / TypeScript)

Vývojové prostredie pre všetky súčasti projektu

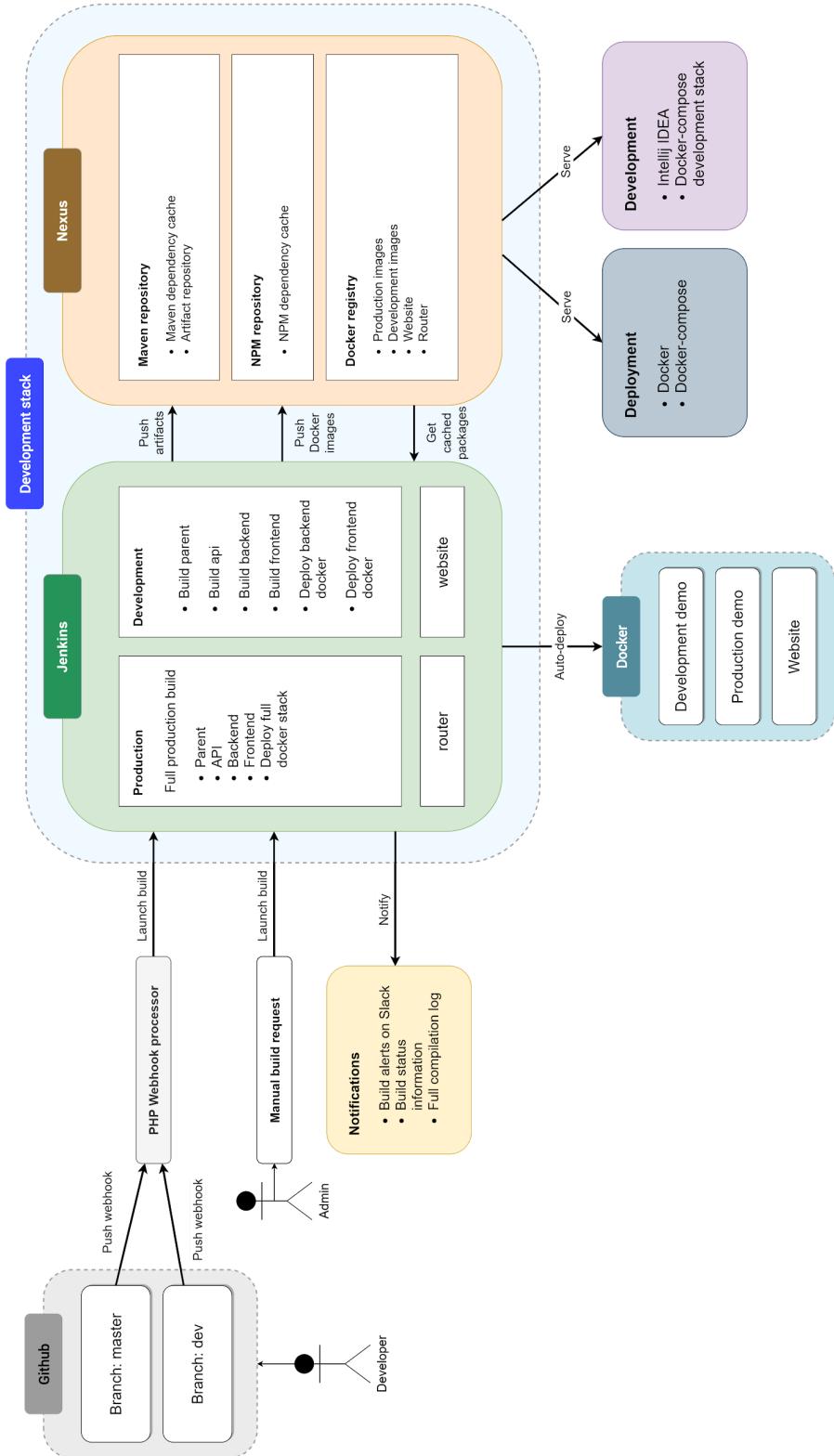
Kvôli komplexnosti projektu a jeho nasadeniu sme si vytvorili automatizované prostredie, ktoré nám uľahčí vývoj, umožní nám odsledovať všetky chyby pri kompliacii kódu ako aj nasadiť aplikáciu bez potreby manuálneho zásahu. Celý kód tohto projektu sa taktiež nachádza na kolaboračnej platforme GitHub.



Využili sme pre to virtuálny stroj, ktorý nám poskytla fakulta v rámci riešenia tohto tímového projektu. Virtuálny stroj je nainštalovaný s operačným systémom Ubuntu 20.04.1 LTS. Na tento stroj sme následne nainštalovali platformu Docker a webový server a proxy Nginx. Na platforme Docker sme nasadili jednotlivé komponenty potrebné pre automatizovaný vývoj a nasadenie ako aj samotné inštancie našej aplikácie. Webový server a proxy nám umožňuje kontrolovať prístup k aplikáciám ako aj smerovať rôzne webové adresy na služby, ktoré využívame. Služba Portainer nám umožňuje sledovať stav a spravovať jednotlivé služby nasadené v rámci platformy Docker.

Medzi služby potrebné pre vývoj patria Jenkins a Nexus. Jenkins je služba ktorá umožňuje spúšťanie automatizovaných úkonov. Využívame ju na spúšťanie kompliacie kódu hned po jeho aktualizácii na platforme GitHub, kde je nás kód uchovávaný. Jenkins sleduje celý priebeh kompliacie a testovania jednotlivých modulov a odosiela notifikácie o úspešnosti kompliacií na náš tímový Slack (náš komunikačný prostriedok). V prípade, že sa v kóde vyskytnú chyby a kompliacia skončí s chybou, vieme na takúto situáciu promptne reagovať a dané chyby opraviť. Nexus je repozitár artefaktov, knižníc, balíkov a Dockerových obrazov. Slúži ako server pre uchovanie rôznych knižníc a balíkov, ktoré využívame v našom projekte.

v pamäti cache, aby ich nebolo potrebné pri každej komplikácii stáhovať z internetu. Okrem toho slúži na uchovanie medzivýsledkov komplikácie a hotových modulov v rámci nášho projektu, čo nám umožňuje zdieľať tieto súbory medzi všetkými programátormi. Jenkins taktiež vytvára na konci úspešnej komplikácie kód obrazy (balíky) pre platformu Docker, ktoré je následne možné nasadiť na akýkoľvek stroj, kde je Docker nainštalovaný, a tieto obrazy sú uložené taktiež na serveri Nexus.



Backend

Backend je súčasť nášho systému, ktorá je zodpovedná za obsluhovanie požiadaviek z používateľského rozhrania. Hlavnými úlohami backendu sú autentifikácia a autorizácia používateľov, správa súborov, správa používateľských účtov, organizácií, tímov a repozitárov.

Táto časť je implementovaná formou API (aplikačného programovacieho rozhrania), ktorého dokumentácia je súčasťou tohto dokumentu a zobrazuje všetky volania, ich vstupy, výstupy a chybové kódy, ktoré môžu požiadavky a odpovede nadobudnúť - Príloha A.

Kedže systém je navrhnutý tak, aby bol modulárny a škálovateľný, jednotlivé časti backendu sú oddelené tak, aby bola táto požiadavka splnená a jednotlivé moduly nemali medzi sebou náväznosti. Jednotlivé moduly systému a ich funkcionality sú opísané v sekcií Moduly backendu.

Použité technológie

V rámci backendu bol použitý programovací jazyk Java, nad ktorým je postavený framework Spring, ktorý slúži na implementovanie robustného API rozhrania. V projekte využívame aj správcu balíkov a verzií Maven. Vďaka tomuto nástroju dokážeme jednoducho pracovať s nadväzujúcimi knižnicami, ktoré sú potrebné v projekte, ako aj jednoducho zadefinovať spôsoby komplikácie našej aplikácie, tvorby artefaktov, ukladanie pomocných súborov do pamäte cache alebo formy testovania aplikácie.

Používané verzie nástrojov:

- Java - OpenJDK 13
- Maven - Apache Maven 3.6.3
- Spring - 2.2.5

Používané verzie knižníc:

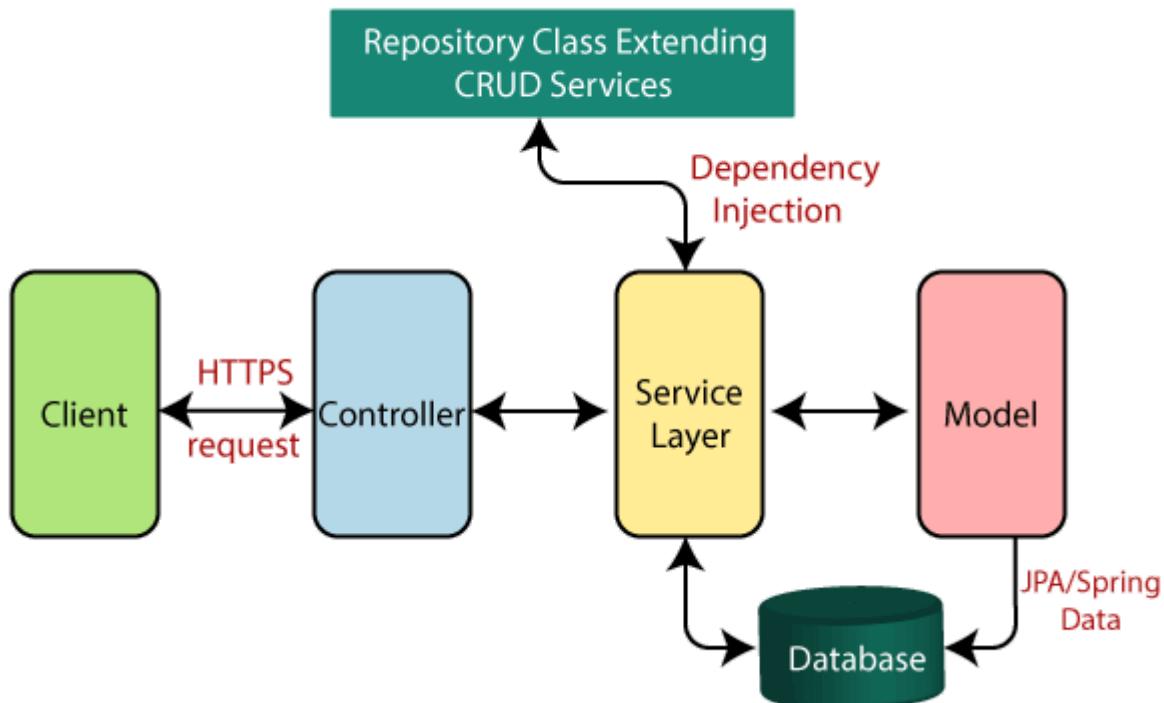
- lombok - 1.18.16
- postgresql - 42.2.5
- liquibase-core - 3.8.5
- modelmapper - 2.3.9
- jackson-jsog - 1.1.1
- java-jwt - 3.11.0

Na vývoj sme používali vývojové štúdio (IDE) IntelliJ Idea (Community edition), keďže tento nástroj poskytuje priamu podporu jazyka Java a poskytuje možnosť nainštalovať si rôzne doplnky pre prácu s frameworkom Spring ako aj knižnicami Lombok či Maven. Okrem užitočných doplnkov a šikovného prostredia poskytuje aj nástroje na odlaďovanie chýb.

Moduly backendu

V tejto časti opíšeme niektoré dôležitejšie moduly a ich úlohy. Všeobecná štruktúra modulu zodpovedá konvenciam vo frameworku Spring, kde sa každý modul skladá z troch časťí (MVC): dátový model, kontrolér a zobrazenie. V realite ide o dátovú vrstvu, servisnú vrstvu a prezentačnú vrstvu.

Spring Boot flow architecture



Zdroj: <https://www.javatpoint.com/spring-boot-architecture>

core

Core modul je hlavný modul našej aplikácie. Tento modul je zodpovedný za spracovanie používateľov, repositárov, organizácií a tímov. Umožňuje sa autentifikovať a autorizovať, vytvárať a spravovať používateľské účty, repositáre, organizácie a tímy. Taktiež je zodpovedný za správu súborov projektov.

parser

Parser modul rieši prácu s kódom ktorý používatelia upravujú v editore aplikácie. Jeho hlavnou úlohou je analýza a validácia tohto kódu.

commons-testing

Tento modul je zodpovedný za spracovanie testov pri ostatných moduloch.

commons-error-handling

Modul commons-error-handling implementuje vlastné error kódy pre aplikáciu. Je knižnica obsahujúca definíciu projektových výnimiek, ktoré sa používajú v každom module. Každá nízko úrovňová výnimka vyhodená knižnicou tretej strany, je zaobalená do jednej z projektových výnimiek definovanej práve v tejto knižnici. Obsahuje tiež konfiguráciu REST API, ktorá každú projektovú výnimku preloží do HTTP odpovede s prislúchajúcim HTTP statusom.

commons-persistence

Obsahuje triedy, ktoré sa opakovane používajú v dátovej vrstve vo viacerých moduloch.

commons-security

commons-security

Je knižnica zodpovedná za zabezpečenie aplikácie. Obsahuje konfiguračné súbory, ktoré nastavujú a dopĺňajú základnú Spring Security vrstvu. Definuje úroveň zabezpečenia pre jednotlivé cesty REST služby, obsahuje v sebe Autentifikačného klienta, pomocou ktorého priamo komunikuje s autentifikačným modulom. Ak je potrebné, aby nový modul podporoval zabezpečenie, stačí importovať knižnicu Commons Security a načítať konfiguračné súbory do kontextu aplikácie.

commons-web

Obsahuje v sebe konfiguračné súbory, ktoré nastavujú webovú vrstvu aplikácie. Importuje Spring Web knižnicu aj s vybranými konfiguračnými súbormi. Konfiguruje Spring Actuator, ktorý umožňuje sledovať základné údaje o aplikácii ako status a rôzne metriky počas behu. Implementuje tiež rôzne triedy, ktoré sa používajú naprieč celým projektom vo webovej vrstve

Aktuálny dátový model



repo

▼ Fields (13)

Name	Type	Settings	References
id	bigint	not_null  PK increment	↳ team.repo_id
uuid	varchar(255)	unique	
name	varchar(50)	not_null unique	
uri	text	not_null	
author	varchar(50)	not_null	
storage_name	varchar(50)	not_null	
file_path	text	not_null	
created	timestamp		
created_by	varchar(255)		
last_modified	timestamp		
last_modified_by	varchar(255)		
favorite	boolean(default: FALSE)		
description	varchar(255)		

▼ Indexes (1)

Columns	Name	Type	Settings
author			

USER

▼ Fields (14)

Name	Type	Settings	References
id	bigint	not_null  PK increment	← user_role.user_id ← team_membership.user_id → organization.owner_id
uuid	varchar(255)	unique	
email	varchar(50)	not_null unique	
username	varchar(50)	not_null unique	
first_name	varchar(50)	not_null	
last_name	varchar(50)	not_null	
password	varchar(255)	not_null	
active	boolean	not_null	
description	varchar(255)		
created	timestamp		
created_by	varchar(255)		
last_modified	timestamp		
last_modified_by	varchar(255)		
organization_id	bigint		→ organization.id

user_role

▼ Fields (3)

Name	Type	Settings	References
id	bigint	not_null  PK increment	
user_id	bigint	not_null	↳ user.id
role	varchar(50)	not_null	

▼ Indexes (1)

Columns	Name	Type	Settings
user_id			

organization

▼ Fields (8)

Name	Type	Settings	References
id	bigint	not_null PK increment	↳ organization_invite.organization_id ↳ team.organization_id — user.organization_id
uuid	varchar(255)	unique	
title	varchar(100)	not_null	
owner_id	bigint	not_null	— user.id
created	timestamp		
created_by	varchar(255)		
last_modified	timestamp		
last_modified_by	varchar(255)		

▼ Indexes (1)

Columns	Name	Type	Settings
owner_id			

organization_invite

▼ Fields (9)

Name	Type	Settings	References
id	bigint	not_null  PK increment	
uuid	varchar(255)	unique	
email	varchar(255)	not_null	
status	varchar(255)		
organization_id	bigint	not_null	↳ organization.id
created	timestamp		
created_by	varchar(255)		
last_modified	timestamp		
last_modified_by	varchar(255)		

▼ Indexes (1)

Columns	Name	Type	Settings
organization_id			

team

▼ Fields (9)

Name	Type	Settings	References
id	bigint	not_null PK increment	← team_membership.team_id
uuid	varchar(255)	unique	
name	varchar(50)	not_null	
organization_id	bigint		→ organization.id
repo_id	bigint		← repo.id
created	timestamp		
created_by	varchar(255)		
last_modified	timestamp		
last_modified_by	varchar(255)		

▼ Indexes (1)

Columns	Name	Type	Settings
repo_id			

team_membership

▼ Fields (9)

Name	Type	Settings	References
id	bigint	not_null  PK increment	
uuid	varchar(255)	unique	
user_id	bigint	not_null	↳ user.id
team_id	bigint	not_null	↳ team.id
permission	varchar(63)	not_null	
created	timestamp		
created_by	varchar(255)		
last_modified	timestamp		
last_modified_by	varchar(255)		

▼ Indexes (2)

Columns	Name	Type	Settings
user_id			
team_id			

Frontend

Frontend je používateľské rozhranie pre tento systém. Sprístupňuje hlavnú stránku projektu ako aj rôzne funkcionality editora. Na pozadí komunikuje s API rozhraním na backende, vďaka ktorému môže registrovať, autentifikovať a autorizovať používateľov, zobrazovať im ich repozitáre, informácie o používateľskom účte, a spravovať organizácie či tímy. Hlavné funkcie editora, ako vizualizácia kódu, oprava syntaxe či validácia písaného kódu sa z väčšej časti realizuje priamo v klientskej aplikácii. Backend slúži hlavne na uchovanie súborov repozitárov.

Samotný frontend sa skladá z viacerých modulov, ako napríklad auth - login, registrácia a autentifikácia používateľa, editor - práca s editorom, home - nastavenia používateľa a repozitárov, app-routing.module - prepája stránky na frontende, a ďalšie. Tieto moduly sú zväčša rozdelené do viacerých komponentov. Každý z komponentov sa skladá z implementácie layoutu v HTML, definovania štýlu v SCSS, testov napísaných v TypeScripte a samotného controlleru taktiež napísanom v TypeScripte. Okrem modulov obsahujúcich komponenty sú na frontende implementované viaceré services, čiže služby, ktoré slúžia na komunikáciu s backendom.

V aktuálnom stave sú implementované nasledovné funkcionality:

- webové rozhranie - využitie frameworku Angular a Typescript
- editor - založený na existujúcom Monaco editore, s pridanou podporou programovacieho jazyka SystemVerilog
- editor - podporuje syntaktické zvýrazňovanie textu, kontrolu syntaktických chýb a automatické dopĺňanie textu
- komunikácia s API rozhraním (backend)
- homepage - úvodná stránka do systému. V prípade, že používateľ nie je prihlásený, je mu ponúknutá možnosť prihlásenia alebo registrácie.
- autorizácia - prihlásenie existujúceho používateľa alebo registrácia nového používateľa
- autentifikácia - používateľ je autentifikovaný pomocou JWT (JSON web token)
- správa používateľa - zahŕňa spravovanie osobných údajov
- správa repozitárov - zahŕňa zobrazenie a spravovanie všetkých osobných repozitárov
- úprava nastavení (osobných a repozitárov) je povolená až po úspešnom prihlásení používateľa

Použité technológie

Klientská časť systému bola implementovaná v jazykoch JavaScript a TypeScript za pomoci framework AngularJS. V rámci tohto frameworku sme využili aj nasledovné dodatočné moduly, ktoré nám sprístupňujú dôležité funkcionality:

- angular/core - 9.1.6
- angular/common - 9.1.6
- angular/material - 9.2.3
- angular/router - 9.1.6
- zip-js - 0.0.2

Na implementovanie samotného prostredia pre písanie kódu sme sa rozhodli namiesto implementovania vlastného editora použiť existujúci open-source editor Monaco. Tento editor disponuje funkcionalitami ako napríklad syntax highlighting, vyznačovanie chýb a upozornenie a pridávanie nových jazykov. Toto nám veľmi uľahčilo prácu a okrem toho je možné tento editor nezávisle od modifikácií, ktoré doňho budú nami implementované, aktualizovať.

Homepage

Domovská stránka predstavuje prvý kontakt s používateľom. Na nej je používateľovi poskytnutá možnosť registrácie, pokiaľ ešte nemá založený účet. V prípade, že používateľ už je registrovaný, môže sa z domovskej stránky priamo prihlásiť. Pri zvolení jednej z dvoch možností, používateľ bude presmerovaný na príslušný formulár, kde vyplní buď svoje prihlásenie údaje (v prípade loginu) alebo svoje osobné údaje potrebné na registráciu. Či už v prípade registrácie, alebo prihlásenia, používateľovi je v oboch prípadoch poskytnuté prejsť na opačnú možnosť (v registračnom formulári prejsť na login a z loginu možnosť prejsť na registráciu). V prípade, že používateľ je prihlásený, domovská stránka ho privít a poskytne mu možnosť prejsť na prehľad svojich repozitárov. Domovská stránka a s ňou súvisiace registrácia a prihlásenie používateľa obsahuje nasledovné komponenty:

- **LandingComponent:** Komponent domovskej stránky, ktorý slúži na to, aby sa zistilo, či je aktuálne prihlásený používateľ alebo nie je. Podľa toho buď privít prihláseného používateľa alebo mu poskytne možnosť prihlásenia a registrácie. Následne presmeruje používateľa na stránku jeho voľby.
- **AuthContainerComponent:** Komponent, ktorý slúži na zobrazenie, formátovanie a následne presmerovanie na stránky registrácie alebo prihlásenia.
- **LoginComponent:** Komponent slúžiaci na prihlásenie používateľa, na prijatie a kontrolu prihlásovacích údajov. V prípade, ak údaje boli nesprávne, respektíve používateľ s danými prihlásovacími údajmi neexistuje, bude vypísaná chyba oznamujúca neúspešné prihlásenie. V opačnom prípade používateľ bude úspešne prihlásený.
- **RegisterComponent:** Komponent zodpovedný za registráciu nového používateľa. V prípade, že boli zadané všetky potrebné údaje v správnom formáte, používateľ bude úspešne zaregistrovaný a systém ho presmeruje na prihlásovaciu stránku, kde sa môže prihlásiť. V opačnom prípade, ak údaje neboli všetky vyplnené alebo nespĺňajú stanovený formát, používateľ bude upozornený.

Prehľad repozitárov

Ďalším prvkom na frontende je prehľad repozitárov prihláseného používateľa. Rozloženie tejto obrazovky sme koncipovali podľa predom pripravených a konzultovaných návrhov (mockupy v prílohe). Vďaka tejto obrazovke má používateľ prístup ku svojim repozitárom, ktoré môže spravovať alebo priamo k nim pristupovať a editovať ich. V prehľade má k dispozícii tie najpodstatnejšie informácie o repozitároch ako je ich názov, cieľ vzdialeneho úložiska, popis repozitáru a pod. Pre tuto funkciaľitu využívané komponenty sú nasledovné:

- **MyReposComponent:** Komponent, ktorý slúži na základnú spravu repozitárov - zobrazenie repozitárov, stránkovanie a umožnenie prístupu ku nim. Umožňuje presmerovanie na ďalšie stránky, ako sú napríklad nastavenia pre daný repozitár.
- **DeleteRepoDialogComponent:** Komponent zodpovedný za samotné odstránenie repozitáru, pričom je používateľ vyzvaný aby permanentne vymazanie repozitáru potvrdil, predtým, ako sa vykoná.

Dashboard organizácie

Kedže manažment organizácií je novou funkcia, zapracovali sme aj na jej prezentačnej stránke. Organizácie vyžadujú omnoho viac nástrojov pre ich manažovanie a práve tie sme sa rozhodli zakomponovať do našej implementácie. Správa tímov a ich členov, repozitárov a ich nastavení je zakomponovaná do časti organizácií. Používateľ má možnosť prehliadať jednotlivé repozitáre a priručovať ich jednotlivým tímom. V časti spravovanie tímov, vidí výpis členov tímu, môže meniť ich poverenia a priručovať jednotlivé repozitáre. Celú správu organizácie uzatvárajú nastavenia, ktoré predstavujú správu organizácie ako celku.

Testovanie

Kedže projekt je rozdelený na dve hlavné časti - backend a frontend, správnu funkcionality týchto častí treba zväčša osobitne testovať. Preto pre obe časti existujú automatické testy, ktoré overia správnosť špecifických požiadaviek. Tieto testy sú v aktuálnom stave, že sa musia opraviť, kedže sme v rámci tohto semestra menili veľa vecí na projekte, aj testy treba meniť a prispôsobiť aktuálnej funkcionality.

Samozrejme hlavný spôsob testovania spočíval v manuálnom testovaní. Toto spočívalo v tom, že vždy keď sa implementovala nová časť, alebo upravila existujúca časť, osoba, ktorá danú zmenu spravila, ju otestovala lokálne u seba. Ak neboli nájdené žiadne chyby, tak konkrétnu zmenu vždy otestoval ešte minimálne jeden ďalší člen z tímu. Ak aj toto bolo v poriadku, mohol sa kód mernúť s hlavným kódom, kde prebla ešte posledná etapa testovania - testovanie cez Jenkins pipelines. Tieto kroky a viaceré etapy testovania nám pomohli minimalizovať chybovosť.

Na backende tiež používame unit testy pomocou Mockito.

Prílohy

Príloha A: API dokumentácia

The screenshot shows the Swagger UI interface for the ASICDE API. At the top, there's a navigation bar with the Swagger logo, a dropdown for selecting the API definition file (.api.json), and a green "Explore" button. Below the header, the title "ASICDE API" is displayed with a version of "1.0.0" and an "OAS3" badge. A link to the ".api.json" file is shown below the title. The main content area contains a brief description: "Cloud Based IDE for SystemVerilog" and links to the "ASICDE Team - Website" and "Send email to ASICDE Team".

Servers

`https://{{environment}}.asicde.org/{{version}} - Public API endpoint`

Computed URL: `https://api.asicde.org/v1`

Server variables

environment: `api` ▾

version: `v1` ▾ Authorize

Auth

`POST /auth/login Authenticate with the API`

Parameters

No parameters

Request body required

`application/json`

Authentication request

Example Value | Schema

```
{ "username": "john.doe", }
```

```
        "password": "supersecret"  
    }
```

Responses

Code	Description	Links
200	Authentication successful	No links

Media type

application/json

Controls Accept header.

Example Value | Schema

```
{  
    "token":  
        "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36P0k6yJV_adQssw5c",  
    "tokenType": "Bearer",  
    "expiration": "2020-11-24T17:01:22.596Z",  
    "user": {  
        "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  
        "username": "john.doe",  
        "email": "john.doe@mail.com",  
        "firstname": "John",  
        "lastname": "Doe",  
        "organization": "3fa85f64-5717-4562-b3fc-2c963f66afa6"  
    }  
}
```

Code	Description	Links												
400	<p>Missing parameters in authentication request</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Password cannot be empty" }</pre> <p>Headers:</p> <table border="1"> <thead> <tr> <th>Name</th><th>Description</th><th>Type</th></tr> </thead> <tbody> <tr> <td>X-Rate-Limit-Limit</td><td>The number of allowed requests in the current period</td><td>integer <i>Example: 10</i></td></tr> <tr> <td>X-Rate-Limit-Remaining</td><td>The number of remaining requests in the current period</td><td>integer <i>Example: 9</i></td></tr> <tr> <td>X-Rate-Limit-Reset</td><td>The number of seconds left in the current period</td><td>integer <i>Example: 59</i></td></tr> </tbody> </table>	Name	Description	Type	X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>	X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer <i>Example: 9</i>	X-Rate-Limit-Reset	The number of seconds left in the current period	integer <i>Example: 59</i>	No links
Name	Description	Type												
X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>												
X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer <i>Example: 9</i>												
X-Rate-Limit-Reset	The number of seconds left in the current period	integer <i>Example: 59</i>												
401	<p>Authentication unsuccessful</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Invalid username or password" }</pre> <p>Headers:</p> <table border="1"> <thead> <tr> <th>Name</th><th>Description</th><th>Type</th></tr> </thead> <tbody> <tr> <td>X-Rate-Limit-Limit</td><td>The number of allowed requests in the current period</td><td>integer <i>Example: 10</i></td></tr> <tr> <td>X-Rate-Limit-Remaining</td><td>The number of remaining requests in the current period</td><td>integer <i>Example: 9</i></td></tr> <tr> <td>X-Rate-Limit-Reset</td><td>The number of seconds left in the current period</td><td>integer <i>Example: 59</i></td></tr> </tbody> </table>	Name	Description	Type	X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>	X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer <i>Example: 9</i>	X-Rate-Limit-Reset	The number of seconds left in the current period	integer <i>Example: 59</i>	No links
Name	Description	Type												
X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>												
X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer <i>Example: 9</i>												
X-Rate-Limit-Reset	The number of seconds left in the current period	integer <i>Example: 59</i>												

Code	Description	Links												
429	<p>Too many requests</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json</div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "Too many requests!" }</pre> <p>Headers:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>X-Rate-Limit-Limit</td> <td>The number of allowed requests in the current period</td> <td>integer Example: 10</td> </tr> <tr> <td>X-Rate-Limit-Remaining</td> <td>The number of remaining requests in the current period</td> <td>integer Example: 0</td> </tr> <tr> <td>X-Rate-Limit-Reset</td> <td>The number of seconds left in the current period</td> <td>integer Example: 37</td> </tr> </tbody> </table>	Name	Description	Type	X-Rate-Limit-Limit	The number of allowed requests in the current period	integer Example: 10	X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer Example: 0	X-Rate-Limit-Reset	The number of seconds left in the current period	integer Example: 37	No links
Name	Description	Type												
X-Rate-Limit-Limit	The number of allowed requests in the current period	integer Example: 10												
X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer Example: 0												
X-Rate-Limit-Reset	The number of seconds left in the current period	integer Example: 37												

GET	/auth/logout De-authenticate current session	
Parameters		Try it out
No parameters		
Responses		
Code	Description	Links
200	OK	No links

GET	/auth/refreshToken Refresh JWT token	
Parameters		Try it out

No parameters

Responses

Code	Description	Links
200	Authentication token refresh successful	No links

Media type

application/json ▾

Controls Accept header.

Example Value | Schema

```
{  
    "token":  
        "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvvaG4gRG9lIiwiawF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c",  
    "TokenType": "Bearer",  
    "expiration": "2020-11-24T17:01:22.608Z"  
}
```

401	Unauthorized request	No links
-----	----------------------	----------

Media type

application/json ▾

Example Value | Schema

```
{  
    "errorCode": 1,  
    "message": "Session already expired"  
}
```

User management

POST /auth/users Create a new user account

Parameters

Try it out

No parameters	Description	Links														
429	<p>Request body <small>required</small></p> <p>Media type User creation request application/json ▾</p> <p>Example Value Schema</p> <pre>{ "username": "john.doe", "password": "supersecret", "email": "john.doe@mail.com", "firstname": "John", "lastname": "Doe" }</pre> <p>Headers:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>X-Rate-Limit-Limit</td> <td>The number of allowed requests in the current period</td> <td>integer <i>Example: 10</i></td> </tr> </tbody> </table> <p>Responses</p> <table border="1"> <thead> <tr> <th>Code</th> <th>Description</th> <th>Type</th> <th>Links</th> </tr> </thead> <tbody> <tr> <td>200</td> <td>X-Rate-Limit-Remaining OK</td> <td>integer <i>Example: 0</i></td> <td>No links</td> </tr> </tbody> </table>	Name	Description	Type	X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>	Code	Description	Type	Links	200	X-Rate-Limit-Remaining OK	integer <i>Example: 0</i>	No links	No links
Name	Description	Type														
X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>														
Code	Description	Type	Links													
200	X-Rate-Limit-Remaining OK	integer <i>Example: 0</i>	No links													
400	<p>Bad user creation request</p> <p>Media type application/json ▾</p> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Password cannot be empty" }</pre> <p>Headers:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>X-Rate-Limit-Limit</td> <td>The number of allowed requests in the current period</td> <td>integer <i>Example: 10</i></td> </tr> <tr> <td>X-Rate-Limit-Remaining</td> <td>The number of remaining requests in the current period</td> <td>integer <i>Example: 9</i></td> </tr> <tr> <td>X-Rate-Limit-Reset</td> <td>The number of seconds left in the current period</td> <td>integer <i>Example: 59</i></td> </tr> </tbody> </table>	Name	Description	Type	X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>	X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer <i>Example: 9</i>	X-Rate-Limit-Reset	The number of seconds left in the current period	integer <i>Example: 59</i>	No links		
Name	Description	Type														
X-Rate-Limit-Limit	The number of allowed requests in the current period	integer <i>Example: 10</i>														
X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer <i>Example: 9</i>														
X-Rate-Limit-Reset	The number of seconds left in the current period	integer <i>Example: 59</i>														

Code	Description	Links
429	Too many requests	No links

Media type

application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "Too many requests!"
}
```

Headers:

Name	Description	Type
X-Rate-Limit-Limit	The number of allowed requests in the current period	integer Example: 10
X-Rate-Limit-Remaining	The number of remaining requests in the current period	integer Example: 0
X-Rate-Limit-Reset	The number of seconds left in the current period	integer Example: 37

PUT /auth/users Update user information 🔒

Parameters Try it out

No parameters

Request body required application/json ▾

User creation request

Example Value | Schema

```
{
  "password": "betterpassword"
}
```

Responses

Code	Description	Links
200	OK	No links
400	Bad user update request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Password is weak" }</pre>	

DELETE	/auth/users Delete user account	
Parameters		Try it out
No parameters		
Responses		
Code	Description	Links
200	OK	No links

Code	Description	Links
401	Unauthorized request	No links

Media type
application/json ▾

Example Value | Schema

```
{  
    "errorCode": 1,  
    "message": "Session already expired"  
}
```


GET /auth/users/{user_uuid} Get user details 

Parameters 

Name	Description
user_uuid * required string(\$uuid) (path)	User identifier string user_uuid - User identifier string

Responses

Code	Description	Links
------	-------------	-------

Code	Description	Links
200	User found	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "user": { "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "username": "john.doe", "email": "john.doe@mail.com", "firstname": "John", "lastname": "Doe", "organization": "3fa85f64-5717-4562-b3fc-2c963f66afa6" } }</pre>	
400	Bad user details request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Invalid user identifier" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	

Code	Description	Links
404	User does not exist	No links

Media type

application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "User does not exist"
}
```

GET	/auth/users/activate/{token} Activate user account	Try it out
Parameters		
Name	Description	
token * required	Activation token	
string (path)	token - Activation token	
Responses		
Code	Description	Links
200	OK	No links

Repository management

GET	/repos Get all user repositories	🔒
-----	----------------------------------	---

Parameters

[Try it out](#)

Name	Description
page integer (query)	Pagination page <input type="text" value="1"/>
size integer (query)	Pagination batch size <input type="text" value="10"/>

Responses

Code	Description	Links
200	Repositories available	No links

Media type

Controls Accept header.

[Example Value](#) | [Schema](#)

```
{
  "pageNum": 1,
  "totalPages": 3,
  "totalItems": 27,
  "data": [
    {
      "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
      "name": "Repository1",
      "url": "git@github.com:ASICDE/documentation.git",
      "archived": false,
      "created": "2020-11-24T17:01:22.651Z",
      "lastModification": "2020-11-24T17:01:22.651Z",
      "team": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
    }
  ]
}
```

Code	Description	Links
400	Paging parameters are invalid	No links
401	Unauthorized request	No links

POST /repos Create new repository 

Parameters 

No parameters

Request body required 

Create a new repository

[Example Value](#) | [Schema](#)

```
{
  "name": "Repository1",
  "url": "git@github.com:ASICDE/documentation.git",
  "archived": false,
  "created": "2020-11-24T17:01:22.660Z",
  "lastModification": "2020-11-24T17:01:22.660Z",
  "team": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "authorization": {
    "type": "plaintext",
    "username": "githubuser",
    "password": "supersecret",
    "sshKey": "string"
  }
}
```

Responses		
Code	Description	Links
200	Repository created successfully	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6" }</pre>	
400	Bad repository create request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Repository already exists with chosen name" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	

GET /repos/{repo_uuid} Get repository

Parameters

		Try it out
--	--	--

Name	Description	
repo_uuid * required string(\$uuid) (path)	Repository identifier string repo_uuid - Repository identifier string	
Responses		
Code	Description	Links
200	Repository exists	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "name": "Repository1", "url": "git@github.com:ASICDE/documentation.git", "archived": false, "created": "2020-11-24T17:01:22.672Z", "lastModification": "2020-11-24T17:01:22.672Z", "team": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "authorization": { "type": "plaintext", "username": "githubuser", "password": "supersecret", "sshKey": "string" } }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	

Code	Description	Links
404	Repository does not exist	No links

Media type

[Example Value](#) | [Schema](#)

```
{
  "errorCode": 2,
  "message": "Repository does not exist"
}
```


PUT	/repos/{repo_uuid} Update repository											
Parameters <div style="float: right; border: 1px solid black; padding: 2px; margin-right: 10px;">Try it out</div> <hr/> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>repo_uuid * required string(\$uuid) (path)</td> <td>Repository identifier string <input style="border: 1px solid black; padding: 2px; width: fit-content;" type="button" value="repo_uuid - Repository identifier string"/></td> </tr> </tbody> </table> <hr/> <div style="display: flex; justify-content: space-between;"> Request body required <input style="border: none; font-size: small;" type="button" value="application/json"/> </div> <hr/> <div style="margin-top: 10px;"> <p>Repo update request</p> <p>Example Value Schema</p> <pre>{ "url": "git@github.com:ASICDE/documentation-new.git" }</pre> </div> <hr/> <div style="margin-top: 10px;"> Responses </div> <table border="1"> <thead> <tr> <th>Code</th> <th>Description</th> <th>Links</th> </tr> </thead> <tbody> <tr> <td>200</td> <td>OK</td> <td>No links</td> </tr> </tbody> </table>			Name	Description	repo_uuid * required string(\$uuid) (path)	Repository identifier string <input style="border: 1px solid black; padding: 2px; width: fit-content;" type="button" value="repo_uuid - Repository identifier string"/>	Code	Description	Links	200	OK	No links
Name	Description											
repo_uuid * required string(\$uuid) (path)	Repository identifier string <input style="border: 1px solid black; padding: 2px; width: fit-content;" type="button" value="repo_uuid - Repository identifier string"/>											
Code	Description	Links										
200	OK	No links										

Code	Description	Links
400	Bad repository update request	No links
	<p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Git URL does not exist" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	
404	Repository does not exist	No links
	<p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "Repository does not exist" }</pre>	

DELETE /repos/{repo_uuid} Delete repository 

Parameters [Try it out](#)

Name	Description
repo_uuid * required string(\$uuid) (path)	Repository identifier string repo_uuid - Repository identifier string

Responses

Code	Description	Links
200	OK	No links
401	Unauthorized request	No links

Media type
application/json ▾

Example Value | Schema

```
{ "errorCode": 1, "message": "Session already expired" }
```

GET /repos/org/ Get all organization repositories 

Parameters Try it out

Name	Description
page integer (query)	Pagination page <input type="text" value="1"/>
size integer (query)	Pagination batch size <input type="text" value="10"/>

Responses

Code	Description	Links
------	-------------	-------

Code	Description	Links
200	<p>Repositories available</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "pageNum": 1, "totalPages": 3, "totalItems": 27, "data": [{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "name": "Repository1", "url": "git@github.com:ASICDE/documentation.git", "archived": false, "created": "2020-11-24T17:01:22.690Z", "lastModification": "2020-11-24T17:01:22.690Z", "team": "3fa85f64-5717-4562-b3fc-2c963f66afa6" }] }</pre>	No links
400	<p>Paging parameters are invalid</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Page index 10 is beyond scale" }</pre>	No links
401	<p>Unauthorized request</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	No links

Code	Description	Links			
404	Not part of organization response	No links			
<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>					
GET	/repos/team/{team_uuid} Get all team repositories				
<p>Parameters</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>Name Description</p> <p>team_uuid * required string(\$uuid) (path) team_uuid - Team identifier string</p> <p>page integer (query) 1</p> <p>size integer (query) 10</p> </div>					
<p>Responses</p> <table border="1"> <thead> <tr> <th>Code</th><th>Description</th><th>Links</th></tr> </thead> </table>			Code	Description	Links
Code	Description	Links			

Code	Description	Links
200	<p>Repositories available</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "pageNum": 1, "totalPages": 3, "totalItems": 27, "data": [{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "name": "Repository1", "url": "git@github.com:ASICDE/documentation.git", "archived": false, "created": "2020-11-24T17:01:22.699Z", "lastModification": "2020-11-24T17:01:22.699Z", "team": "3fa85f64-5717-4562-b3fc-2c963f66afa6" }] }</pre>	No links
400	<p>Paging parameters are invalid</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Page index 10 is beyond scale" }</pre>	No links
401	<p>Unauthorized request</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	No links

Code	Description	Links
404	Team does not exist	No links

Media type

application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "Team does not exist"
}
```

Organization management

GET
/org Get organization information
🔒

Parameters
Try it out

No parameters

Responses

Code	Description	Links
200	Organization exists	No links

Media type

application/json ▾

Controls Accept header.

Example Value | Schema

```
{
  "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "name": "Acme",
  "owner": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}
```

Code	Description	Links
401	Unauthorized request	No links
404	Not part of organization response	No links

POST /org Create a new organization 

Parameters 

No parameters

Request body required 

Organization creation request

[Example Value](#) | [Schema](#)

```
{
  "name": "Acme"
}
```

Responses

Code	Description	Links
200	Organization created successfully	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6" }</pre>	
400	Bad organization create request	No links

401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Invalid organization name" }</pre>	

PUT /org Update organization owner 

Parameters	Try it out
No parameters	
Request body <small>required</small>	application/json ▾
Organization team member update request	

[Example Value](#) | [Schema](#)

```
{  
    "owner": "3fa85f64-5717-4562-b3fc-2c963f66afa6"  
}
```

Responses

Code	Description	Links
200	OK	<i>No links</i>
400	Bad organization update request	<i>No links</i>
Media type		
application/json ▾		
Example Value Schema		
{ "errorCode": 0, "message": "User does not exist" }		
401	Unauthorized request	<i>No links</i>
Media type		
application/json ▾		
Example Value Schema		
{ "errorCode": 1, "message": "Session already expired" }		
404	Not part of organization response	<i>No links</i>
Media type		
application/json ▾		
Example Value Schema		
{ "errorCode": 2, "message": "You are not part of any organization" }		

DELETE /org Delete organization 

Parameters

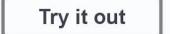
No parameters 

Responses

Code	Description	Links
200	OK	No links
401	Unauthorized request	No links
	Media type application/json ▾	
	Example Value Schema	
	{ "errorCode": 1, "message": "Session already expired" }	
404	Not part of organization response	No links
	Media type application/json ▾	
	Example Value Schema	
	{ "errorCode": 2, "message": "You are not part of any organization" }	

GET /org/settings Get organization settings 

Parameters

No parameters 

Responses

Code	Description	Links
200	Organization exists	No links
	<p>Media type</p> <div><p>application/json</p><p>Controls Accept header.</p></div> <p>Example Value Schema</p> <pre>{ "name_policy": "username", "reader_by_default": true, "authorization": { "type": "plaintext", "username": "githubuser", "password": "supersecret", "sshKey": "string" } }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div><p>application/json</p></div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	
404	Not part of organization response	No links
	<p>Media type</p> <div><p>application/json</p></div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>	

PUT

/org/settings Change organization settings



Parameters

Try it out

No parameters

Request body required

application/json

Organization settings update request

Example Value | Schema

```
{  
  "name_policy": "fullname"  
}
```

Responses

Code	Description	Links
200	OK	No links
400	Bad organization settings update request	No links
401	Unauthorized request	No links

Media type

application/json

Example Value | Schema

```
{  
  "errorCode": 0,  
  "message": "Invalid name policy value"  
}
```

Media type

application/json

Example Value | Schema

```
{  
  "errorCode": 1,  
  "message": "Session already expired"  
}
```

Code	Description	Links
404	Not part of organization response	No links

Media type

application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "You are not part of any organization"
}
```

Organization member management

GET	/org/members	Get organization members	Try it out
Parameters			
Name	Description		
page integer (query)	Pagination page	1	
size integer (query)	Pagination batch size	10	
Responses			
Code	Description	Links	

Code	Description	Links
200	Organization exists	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "pageNum": 1, "totalPages": 3, "totalItems": 27, "data": [{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "username": "john.doe", "email": "john.doe@mail.com", "firstname": "John", "lastname": "Doe", "role": "member" }] }</pre>	
400	Paging parameters are invalid	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Page index 10 is beyond scale" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	

Code	Description	Links
404	Not part of organization response	No links
	<p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>	

POST /org/members Invite organization member

Parameters	Try it out
No parameters	
Request body <small>required</small>	application/json ▾
Organization invitation request	
<p>Example Value Schema</p> <pre>{ "invite": "alice.gordon@mail.com" }</pre>	

Responses		
Code	Description	Links
200	OK	No links

Code	Description	Links
400	Bad organization invite request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Invalid email address" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	
404	Not part of organization response	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>	

PUT /org/members/{member_uuid} Update organization member role 

Parameters	Try it out
<p>Name</p> <p>member_uuid * required string(\$uuid) (path)</p>	<input type="button" value="Try it out"/>
<p>member_uuid - User identifier string</p>	

Request body required

application/json ▾

Organization team member update request

Example Value | Schema

```
{  
  "role": "admin"  
}
```

Responses

Code	Description	Links
200	OK	No links
400	Bad organization member update request	No links
401	Unauthorized request	No links

Media type

application/json ▾

Example Value | Schema

```
{  
  "errorCode": 0,  
  "message": "Invalid role"  
}
```

Media type

application/json ▾

Example Value | Schema

```
{  
  "errorCode": 1,  
  "message": "Session already expired"  
}
```

Code	Description	Links				
404	Not part of organization response	No links				
<p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div>						
<p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>						
<p>DELETE /org/members/{member_uuid} Remove organization member </p>						
<p>Parameters</p> <table border="1"> <thead> <tr> <th>Name</th><th>Description</th></tr> </thead> <tbody> <tr> <td>member_uuid * required string(\$uuid) (path)</td><td>User identifier string member_uuid - User identifier string</td></tr> </tbody> </table>			Name	Description	member_uuid * required string(\$uuid) (path)	User identifier string member_uuid - User identifier string
Name	Description					
member_uuid * required string(\$uuid) (path)	User identifier string member_uuid - User identifier string					
<p>Responses</p>						
Code	Description	Links				
200	OK	No links				
401	Unauthorized request	No links				
<p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json ▾</div>						
<p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>						

Code	Description	Links
404	Not part of organization response	No links

Media type

application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "You are not part of any organization"
}
```

Organization team management

GET
/org/teams Get all teams in organization


Parameters


Name	Description
page integer (query)	Pagination page <input type="text" value="1"/>
size integer (query)	Pagination batch size <input type="text" value="10"/>

Responses

Code	Description	Links

Code	Description	Links
200	Organization exists	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "pageNum": 1, "totalPages": 3, "totalItems": 27, "data": [{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "name": "Team1", "owner": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "member": true }] }</pre>	
400	Paging parameters are invalid	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Page index 10 is beyond scale" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	

Code	Description	Links
404	Not part of organization response	No links

Media type

application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "You are not part of any organization"
}
```

POST /org/teams Create a new team 

Parameters Try it out

No parameters

Request body required application/json ▾

Team creation request

Example Value | Schema

```
{
  "name": "Team1"
}
```

Responses

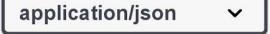
Code	Description	Links

Code	Description	Links
200	Team created successfully	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6" }</pre>	
400	Bad team create request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Invalid team name" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	
404	Not part of organization response	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>	

PUT /org/teams Update team name 

Parameters

No parameters

Request body required 

Organization team update request

[Example Value](#) | [Schema](#)

```
{ "name": "Team2" }
```

Responses

Code	Description	Links
200	OK	<i>No links</i>
400	Bad organization team update request	<i>No links</i>
401	Unauthorized request	<i>No links</i>

400 Bad organization team update request

Media type 

[Example Value](#) | [Schema](#)

```
{ "errorCode": 0, "message": "Invalid team name" }
```

401 Unauthorized request

Media type 

[Example Value](#) | [Schema](#)

```
{ "errorCode": 1, "message": "Session already expired" }
```

Code	Description	Links
404	Not part of organization response	No links

Media type
 application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "You are not part of any organization"
}
```

GET /org/teams/{team_uuid} Get team details 

Parameters 

Name	Description
team_uuid * required string(\$uuid) (path)	Team identifier string team_uuid - Team identifier string

Responses

Code	Description	Links
200	Organization exists	No links

Media type
 application/json ▾

Controls Accept header.

Example Value | Schema

```
{
  "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "name": "Team1",
  "owner": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "member": true
}
```

Code	Description	Links
401	Unauthorized request	No links
404	<p>Not part of organization response</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	No links

DELETE /org/teams/{team_uuid} Remove a team

Parameters Try it out

Name	Description
team_uuid * required string(\$uuid) (path)	Team identifier string

team_uuid - Team identifier string

Responses

Code	Description	Links
200	OK	No links

Code	Description	Links
401	Unauthorized request	No links
404	<p>Not part of organization response</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	No links

GET
/org/teams/{team_uuid}/members Get all team members


Parameters

Parameters
[Try it out](#)

Name	Description
team_uuid * required	Team identifier string
string(\$uuid) (path)	team_uuid - Team identifier string
page	Pagination page
integer (query)	1
size	Pagination batch size
integer (query)	10

Responses		
Code	Description	Links
200	Organization exists	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "pageNum": 1, "totalPages": 3, "totalItems": 27, "data": [{ "uuid": "3fa85f64-5717-4562-b3fc-2c963f66afa6", "username": "john.doe", "email": "john.doe@mail.com", "firstname": "John", "lastname": "Doe", "role": null }] }</pre>	
400	Paging parameters are invalid	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Page index 10 is beyond scale" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▼</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	

Code	Description	Links
404	Not part of organization response	No links

Media type

application/json ▾

Example Value | Schema

```
{
  "errorCode": 2,
  "message": "You are not part of any organization"
}
```

POST /org/teams/{team_uuid}/members Add new team member 

Parameters Try it out

Name	Description
team_uuid * required string(\$uuid) (path)	Team identifier string team_uuid - Team identifier string

Request body required application/json ▾

Organization team member add request

Example Value | Schema

```
{
  "uuid": "ad54a995-b35b-4a42-9353-629cb205d12b",
  "role": "admin"
}
```

Responses

Code	Description	Links
200	OK	No links

Code	Description	Links
400	Bad organization team member add request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 0, "message": "Invalid user identifier" }</pre>	
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	
404	Not part of organization response	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div> <p>Example Value Schema</p> <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>	

PUT /org/teams/{team_uuid}/members/{member_uuid} Update team member role 

Parameters	Try it out
<p>Name</p> <p>team_uuid * required string(\$uuid) (path)</p>	team_uuid - Team identifier string

Name	Description
member_uuid * required string(\$uuid) (path)	User identifier string member_uuid - User identifier string

Request body required

application/json ▾

Organization team member update request

Example Value | Schema

```
{  
  "role": "contributor"  
}
```

Responses

Code	Description	Links
200	OK	No links
400	Bad organization team member update request	No links

Media type

application/json ▾

Example Value | Schema

```
{  
  "errorCode": 0,  
  "message": "Invalid role"  
}
```

Code	Description	Links
401	Unauthorized request	No links
404	<p>Not part of organization response</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">application/json</div> <p>Example Value Schema</p> <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	No links

DELETE /org/teams/{team_uuid}/members/{member_uuid} Remove a team member 

Parameters
[Try it out](#)

Name	Description
team_uuid * required string(\$uuid) (path)	Team identifier string team_uuid - Team identifier string
member_uuid * required string(\$uuid) (path)	User identifier string member_uuid - User identifier string

Responses

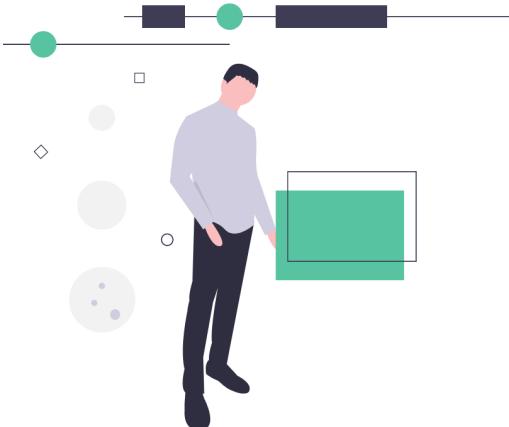
Code	Description	Links
200	OK	No links
401	Unauthorized request	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div>	
	Example Value Schema <pre>{ "errorCode": 1, "message": "Session already expired" }</pre>	
404	Not part of organization response	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json ▾</div>	
	Example Value Schema <pre>{ "errorCode": 2, "message": "You are not part of any organization" }</pre>	

Schemas >

VALID {…}

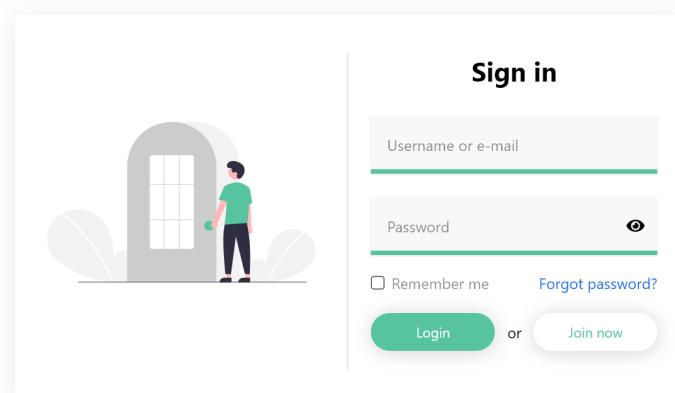
Príloha B: Frontend Mockup

Landing page



The landing page features a header with the logo "ASICDE" and navigation links for About, Features, Pricing, Contact, and TOS. Below the header is a large image of a man in a grey shirt and dark pants interacting with a green rectangular interface element. To the left of the man are several small icons: a green circle, a black rectangle, a green diamond, a grey circle, a white square, and a grey circle with dots. To the right of the man is a call-to-action section with the text "Your IDE of choice for SystemVerilog", a "Try it now, for free!" button, and two buttons: "Join now" and "Log in".

Login



The login page features a header with the logo "ASICDE" and navigation links for About, Features, Pricing, Contact, and TOS. The main content area contains a "Sign in" form with fields for "Username or e-mail" and "Password", a "Remember me" checkbox, a "Forgot password?" link, and "Login" and "Join now" buttons. To the left of the form is a background illustration of a person in a green shirt standing at the entrance of a building.

Registrácia

ASICDE

About | Features | Pricing | Contact | TOS



Join ASICDE

Full name

Username

E-mail

Password

Join now or Log in

Homepage

ASICDE

+ Create repository

My repositories | My account | Log out

Hi Karolína,

Here are your repositories

Repository 1 active

The description goes here

Synchronized with git: <https://www.github.com/katrn/repo1.git>

Repository 2 active

The description goes here

Synchronized with git: <https://www.github.com/katrn/repo2.git>

Repository 3 active

The description goes here

Local

Repository 4 active

The description goes here

Local

Repository 5 active

The description goes here

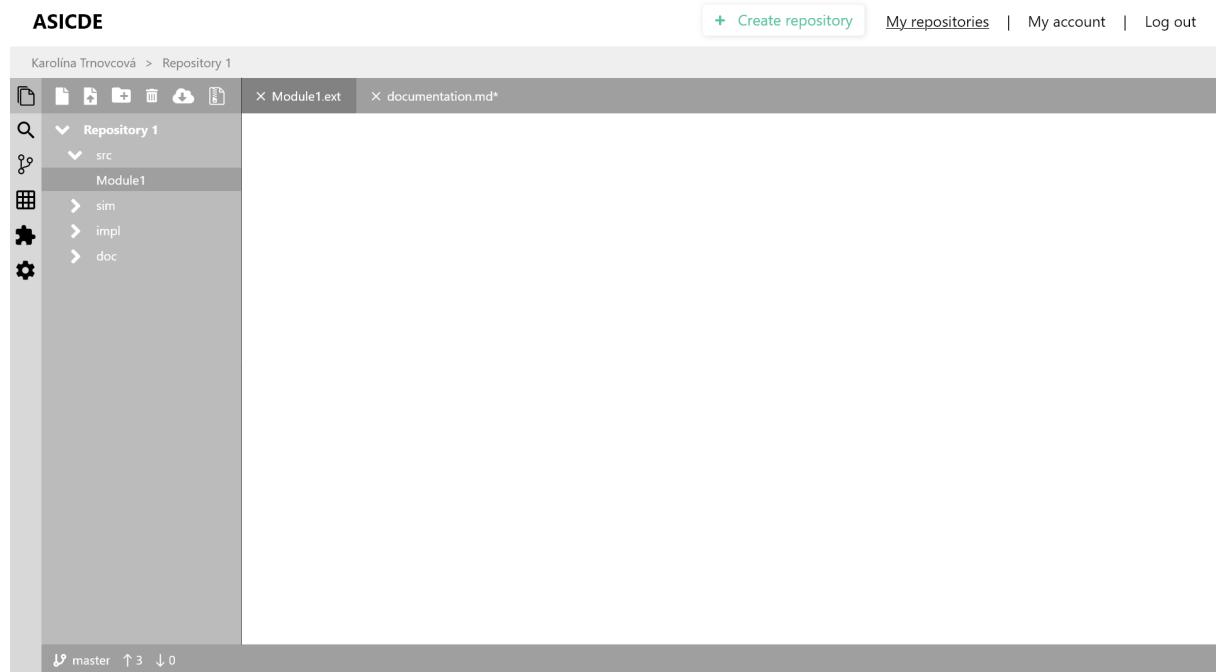
Local

Repository 6 archived

The description goes here

Synchronized with git: <https://www.bibucket.org/repo6.git>

Editor



Organization Members

The screenshot shows the ACME organization members overview page with the following elements:

- Header:** ASICDE: ACME.org, My organization, My repositories, My account, Log out.
- Section:** ACME org overview.
- Buttons:** Repositories, Teams, Members (highlighted).
- Section:** Organization members.
- Table:** A list of organization members with columns for Name, Role, and Teams.

Name	Role	Teams
Joe Blow j.blow	Owner	3 teams
Steve Blow s.blow	Member	3 teams
Billy Blow b.blow	Member	1 team
Jane Blow j.blowz	Member	4 teams

Team Management

ASICDE: ACME org > Team \$Name

[My organization](#) | [My repositories](#) | [My account](#) | [Log out](#)

Team \$Name

Team members

	Tadeáš Drahovský Team Leader	
	Karolina Trnovecová Member	

Team repositories

	Repository 1 <small>active</small> The description goes here Synchronized with git: https://www.github.com/katrn/repo1.git	
	Repository 2 <small>archived</small> The description goes here	

[+ Create a new repository](#)

Príloha C: Frontend screenshoty

Account settings

Account settings

Your basic information

First name: tester

Last name: tester

Email: tester05@gmail.com

Description: Hi, I'm tester 0505

Change your password

Old password: (empty)

New password: (empty)

New password again: (empty)

[Change your password](#)

Account operations

[Delete your account](#)

Organization operations

[Leave your organization](#)

[Delete your organization](#)

[Update information](#)

Modal confirmation

Account settings

Your basic information

First name: tester

Last name: tester

Email: tester05@gmail.com

Description: Hi, I'm tester 0505

Change your password

Old password: (empty)

New password: (empty)

New password again: (empty)

[Change your password](#)

Account operations

[Delete your account](#)

Organization operations

[Leave your organization](#)

[Delete your organization](#)

Delete account?

Are you sure you want to delete your account? This action cannot be undone.

[Cancel](#) [Delete](#)

[Update information](#)