

# Zápis z 11. stretnutia tímu č. 5

**Dátum:** 15.2.2012  
**Miestnosť:** softvérové štúdio (D 003)  
**Prítomní:** Vedúci: Ing. Ivan Kapustík  
Ing. Marián Lekavý, PhD.  
Študenti: Bimbo Miroslav, Bc.  
Boleček Tomáš, Bc.  
Sedláček Andrej, Bc.  
Šimko Ivan, Bc.  
Iní: členovia tímu 17  
**Téma:** Prvé stretnutie po prázdninách, rozdelenie úloh, predstavenie hrubého plánu na semester  
**Vypracoval:** Miroslav Bimbo

## Opis stretnutia

- Obsahom stretnutia bolo rozdelenie úloh medzi jednotlivých členov tímov. Vychádzalo sa z existujúceho product backlogu a nových požiadaviek.
- Okolo polovice semestra už treba mať hotový produkt tak, aby fungoval (alfa verziu), neskôr sa ešte môže doladovať
- Finálna prezentácia bude koncom júna
- Súťaže
  - Blíži sa IIT.SRC,
    - kde sa každý rok prezentuje Robocup, treba si pripraviť plagát.
    - Vlni bol Robocup zvlášť od iných, poster pre oba tímy, premietalo sa cez dataprojektor, je zaujímavé ukazovať aj rôzne nevydarené situácie a ukážky hry zo sveta z youtube.
  - Súťaž medzi tímami
    - bude koncom mája, zvyčajne v piatok, aby sa neprekrývala so skúškami,
    - zúčastňujú sa na nej zvyčajne aj bakalári, diplomanti a minuloroční študenti.
    - Na stránke FIIT Robocupu je odkaz aj na súťaž.
    - Tímy by si mali na úrovni zručností a stratégie konkurovať (napriek tomu, že ich riešenia musia byť vzájomne kompatibilné).
    - Súťaž sa môže uskutočniť napr. v TPC, toto sa dorieši neskôr.
- Úlohy, ktoré treba riešiť:
  - Predikcia
    - Otázka, či by ju nebolo dobré riešiť prostredníctvom knižnice, ktorú používa RC server. Takéto riešenie je možné skúsiť. Môže to byť súčasťou napr. predikcie lopty.
    - Odhad napr. oblúčika – zaujímavá je len horizontálna zložka, dá sa dopočítať prostredníctvom miery tlmenia.
  - Pohyby
    - riešiť finalizáciu pohybov, aby sa pohyb vedel bezpečne zastaviť, aj ak je hráč rozbehnutý. Čím rýchlejšie beží, tým horšie sa zastavuje. Pôvodná myšlienka je, že vykonávanie pohybu sa dostane do finalizácie, v ktorej sa pohyb korektne ukončí, napríklad aj spomalením. Treba si pozrieť kód, v XML je to urobené tak, aby to fungovalo.

- Optimalizácia pohybov je nekonečná úloha, pohyby sa dajú vždy ešte vylepšovať
  - Plánovanie a spájanie pohybov
    - Prispôbenie pohybov tak, aby sa dali zreťaziť,
    - zistiť, prečo sú zreťazené pohyby menej stabilné (pravdepodobne niekde uchádza pár taktov)
    - Spájanie pohybov, zatiaľ aspoň na tej úrovni, aby sa hráč vedel dostať z bodu A do bodu B a správne sa natočiť.
    - Riešenie vhodnosti nahrávky – odhad pravdepodobnosti úspešnosti strely, nahrávky. Zaujímavé je zistiť, či hráč leží, alebo stojí, podľa toho sa dá určiť kam mu nahráť, a či vôbec (toto sa dá použiť aj pri zisťovaní hernej situácie a pod...)
    - Treba vedieť spraviť preplánovanie v strede vyššieho pohybu,
  - Anotácie
    - Každá anotácia má mať inú predpokladku, je teda vhodné, aby jeden pohyb mal viacero anotácií, spájanie viacero anotácií do jednej je zatiaľ zbytočná optimalizácia.
  - Paralelné spracovanie
    - Táto úloha sa cez prázdniny nikam neposunula, preto je určená ako neprioritná, ak by sa to niekam predsalen začalo posúvať, je možné ju uznať ako úlohu za body.
  - Testovací framework
    - aktuálny stav: vie komunikovať s hráčom (aký pohyb hráč vykonáva a aký má vykonávať), zapnúť hráča, server. Boli vytvorené niektoré základné testy pohybov.
    - Čo treba: vytvoriť model frameworku na viacerých úrovniach: komunikácia, vnútorný svet, atď... tj. zistiť, čo všetko ešte chýba, čo všetko framework „vidí“ a čo všetko by mal „vidieť“, akú komunikáciu s hráčom bude potrebovať (aké informácie môže potrebovať pri rôznych pohyboch?),
    - Aké testy na vyhodnocovanie budú potrebné? (koľko to pohybu trvalo, aký bol stabilný (penalizácia za pád) , zaujímavé môže byť aj info, ako je ho možné spájať, aká je stratégia, ...). Niekedy treba dopočítať, napr. koľko by to danému pohybu trvalo, kým by prišiel do cieľa... ?
    - Treba si pozrieť v čom sa sťažujú a nakódovať to.
  - Vytvoriť spoločné knižnice pre hráča a framework
    - Tie časti kódu, ktoré používajú oba programy
  - Vylepšiť GUI pre framework
    - Napr. tlačidlo, ktoré povie o hráčovi všetky dostupné informácie
    - V návrhu radšej viac obrázkov
  - Refactoring
    - identifikovať časti kódu, kde sú problémy, riešiť ich
  - Vyhodnocovanie hernej situácie
    - Kto drží loptu, kto je v útoku, v obrane a pod... Iba určenie pozície lopty nie je postačujúce na určenie, komu patrí (treba brať v úvahu napr. či hráč neleží, kam lopta smeruje a pod...)
- Každá úloha musí mať už v prvý týždeň hotový minimálne návrh, vhodné je už aj začať s implementáciou.
- Čo najskôr je potrebné poslať rozvrhy

## Úlohy na ďalší týždeň:

ID	Člen tímu	Popis úlohy
sp5.1	Miroslav Bimbo	Vylepšenie autoamtického anotovania
sp5.2	Tomáš Boleček	Vylepšenie GUI testovacieho frameworku
sp5.3	Tomáš Boleček	Refactoring testovacieho frameworku
sp5.4	Tomáš Boleček	Vytvorenie spoločných knižníc pre testovací framework a hráča
sp5.5	Andrej Sedláček	Optimalizácia pohybov
sp5.6	Ivan Šimko	Opis a upgrade testovacieho frameworku

Podrobnejšie informácie:

sp5.1 – Vylepšenie autoamtického anotovania – rozšírenie hodnôt predpodmienky a výstupné podmienky (hodnota a odchýlka), vytvorenie viacerých anotácií pre každý pohyb, implementácia

sp5.2 - Vylepšenie GUI testovacieho frameworku – vytvoriť návrhy, zistiť čo všetko bude potrebné, implementácia

sp5.3 - Refactoring testovacieho frameworku – zistiť kde v kóde sú problémy, opraviť, komentáre.

sp5.4 - Vytvorenie spoločných knižníc pre testovací framework a hráča – zistiť, kde používa hráč a framework rovnaké časti kódu a oddeliť ich zvlášť do knižnice používanej oboma programami.

sp5.5 - Optimalizácia pohybov – zistiť, o čom by mohol pri danom pohybe hráč informovať framework, dohodnúť fungovanie finalizácie (s druhým tímom), optimalizovať pohyby.

sp5.6 – Opis a upgrade testovacieho frameworku – zistiť, aký je model testovacieho frameworku, a aký by mal byť - akú komunikáciu bude potrebovať, aké budú potrebné testy, čo by mal vidieť. Implementácia.